Fouille de séquences d'images médicales. Application en chirurgie mini-invasive augmentée

Mohammed Zakarya Droueche

N° d'ordre : 2012telb0245

Sous le sceau de l'Université européenne de Bretagne

Télécom Bretagne

En habilitation conjointe avec l'Université de Rennes 1

Ecole Doctorale – MATISSE

Fouille de séquences d'images médicales. Application en chirurgie mini-invasive augmentée

---

Thèse de Doctorat

Mention : Informatique

Présentée par **Mohammed Zakarya Droueche**

Département : Image et Traitement de l'Information

Laboratoire de Traitement de l'Information Médicale, LaTIM UMR1101 INSERM

Directeur de thèse : Christian Roux

Soutenue le 10 décembre 2012

Jury :

Mme Marie-Odile Berger, Directrice de l'équipe MAGRIT, LORIA (UMR 7503), INRIA Nancy Grand Est (Rapporteur)
M. Philippe Poignet, Professeur, Université de Montpellier II (Rapporteur)
M. Christian Roux, Professeur, Télécom Bretagne (Directeur de thèse)
Mme Mireille Garreau, Professeur, Université de Rennes I (Examinatrice)
Mme Béatrice Cochener, Professeur (PU-PH), Université de Bretagne Occidentale (Examinatrice)
M. Guy Cazuguel, Directeur d'Etudes, Télécom Bretagne (Examinateur)
M. Mathieu Lamard, Ingénieur de recherche, Université de Bretagne Occidentale (Invité)
M. Gwénolé Quellec, Chargé de Recherche INSERM, LaTIM (Invité)

Résumé

DANS cette thèse, nous nous intéressons à l'aide à la décision lors d'interventions chirurgicales. Dans ce but, nous proposons d'utiliser des enregistrements vidéos acquis lors d'interventions chirurgicales antérieures, vidéos numérisées et archivées dans des dossiers d'intervention, contenant toutes les informations relatives à leur déroulement. Au cours de l'opération, le chirurgien ne peut pas consulter lui-même des dossiers et vidéos déjà archivées car il est totalement concentré sur l'acte ; par contre des outils d'analyse automatique en temps réel des images acquises en cours d'opération pourraient permettre cette utilisation de séquences déjà archivées, avec comme applications directes : des alertes en cas de problème, des informations sur les suites de tel ou tel geste dans des situations opératoires voisines (opération, caractéristiques patient, etc ...), des conseils sur les décisions. Notre objectif est donc de développer des méthodes permettant de sélectionner dans des archives des vidéos similaires à la vidéo proposée en requête. Nous nous appuyons pour cela sur la recherche de vidéos par le contenu (CBVR : *Content Based Video Retrieval*) et le raisonnement à base de cas (CBR : *Case Based Reasoning*). Les méthodes sont évaluées sur trois bases de données. Les deux premières bases de données étudiées sont des bases réalisées en chirurgie ophtalmologique, en collaboration avec le service d'ophtalmologie du CHRU de Brest : une base de chirurgie de pelage de membrane de la rétine et une base de chirurgie de la cataracte. La troisième base est la base de clips vidéo Hollywood, utilisée pour montrer la généricité des méthodes proposées.

Pour caractériser les vidéos, nous proposons trois méthodes originales d'indexation à partir du domaine compressé : 1) une première méthode consiste à caractériser globalement la vidéo en utilisant des histogrammes de directions de mouvement, 2) une deuxième méthodes est basée sur une segmentation spatio-temporelle et sur le suivi des régions entre deux images I, pour construire une signature décrivant la trajectoire des régions identifiées comme les plus importantes visuellement, 3) la troisième méthode est une variante de la deuxième méthode : afin de réduire la perte d'information engendrée en utilisant uniquement les images I, nous avons construit un résumé de la vidéo basé sur une sélection des Group Of Pictures (groupes d'images définis dans la norme de compression). Une des originalités de ces trois méthodes est d'utiliser les données vidéos dans le domaine compressé. Ce choix nous permet d'accéder à des éléments caractérisant les vidéos d'une manière rapide et efficace, sans devoir passer par la reconstruction totale du flux vidéo à partir du flux compressé.

Une fois les vidéos caractérisées, la recherche s'effectue en calculant, au sens d'une métrique donnée, la distance entre la signature des vidéos requête et les signatures des vidéos de la base. Ce calcul permet de sélectionner des vidéos en réponse à la requête en dehors de toute signification sémantique. Nous avons proposé trois méthodes de calcul de distance. Tout d'abord, l'algorithme classique "alignement dynamique temporel", ou "Dynamic Time Warping (DTW)" : il permet d'obtenir efficacement la distance entre deux séquences d'images. Cet algorithme est à l'origine de l'algorithme rapide FDTW que nous utilisons pour comparer les signatures issues de la première méthode. Nous présentons ensuite la distance EMD (Earth Mover's Distance), qui nous a conduit à la distance EFDTW (Extented Fast

Dynamic Time Warping), en la combinant avec l'algorithme FDTW (Fast Dynamic Time Warping). Nous l'utilisons pour comparer les signatures de la deuxième et de la troisième méthode, basées sur la trajectoire des régions. Pour améliorer le résultat de retrouvaille, nous introduisons une technique d'optimisation pour le calcul de la distance entre signatures, en utilisant les algorithmes génétiques.

Les résultats que nous obtenons pour les deux bases de données médicales que nous étudions sont très encourageants. Ainsi, la précision moyenne pour une fenêtre de cinq cas atteint 79% (4 vidéos sont similaires à la vidéo requête) pour la base de pelage de membrane, et 72,69% pour la base de la cataracte (3 à 4 vidéos sont similaires à la vidéo requête).

Mots clés : indexation, recherche de vidéo par le contenu, suivi de régions, alignement dynamique temporel.

Abstract

In this thesis, we are interested in computer-aided ophthalmic surgery. In this goal, we propose to use surgery videos already stored in database and associated with contextual information (data patients, diagnostics ... etc). During the surgery, the surgeon is focused on his task. We try to improve the surgical procedures by proposing a system able, at any time, to guide the surgery steps by generating surgical warnings or recommendations if the current surgery shares signs of complications with already stored videos. Our goal is to develop methods and a system to select in the databases videos similar to a video stream captured by a digital camera monitoring the surgery (query). Our work will therefore implement methods related to Content Based Video Retrieval (CBVR) and Case-Based Reasoning (CBR). The methods are evaluated on three databases. The first two databases are collected at Brest University Hospital (France) : the epiretinal membrane surgery dataset and the cataract surgery dataset. Third, in order to assess its generality, the system is applied to a large dataset of movie clips (Holywood) with classified human actions.

To caracterize our videos, we proposed three original indexing methods derived from the compressed "MPEG-4 AVC/H.264" video stream. 1) A global method is based on motion histogram created for every frame of a compressed video sequence to extract motion direction and intensity statistics. 2) A local method combine segmentation and tracking to extract région displacements between consecutive I-frames and therefore characterize région trajectories. 3) To reduce the loss of information caused by using only the I-frames, we constructed a summary of each video based on a selection of the Group Of Pictures (GOP defined in the standard of compression). An originality of these methods comes from the use of the compressed domain, they not rely on standard methods, such as the optical flow, to characterize motion in videos. Instead, motion is directly extracted from the compressed MPEG stream. The goal is to provide a fast video characterization.

Once videos are characterized, search is made by computing, within the meaning of a given metric, the distance between the signature of the query video and the signature of videos in the database. This computing can select videos as answer to the query without any semantic meaning. For this we use three methods. DTW (Dynamic Time Warping) provides an effective distance between two sequences of images. This algorithm is at the origin of the fast algorithm (FDTW) that we use to compare signatures in the first method. To compare signatures resulting from approach based on région motion trajectories, we propose to use a combination of FDTW and EMD (Earth Mover's Distance). The proposed extension of FDTW is referred to as EFDTW. To improve the retrieval result, we introduce an optimization process for computing distances between signature, by using genetic algorithms.

The results obtained on the two medical databases are satisfactory. Thus, the mean precision at five reaches 79% (4 videos similar to the query video) on the epiretinal membrane

surgery dataset and 72,69% (3 to 4 videos similar to the query video) on the cataract surgery dataset.

Keywords : indexing, content-based video retrieval, régions traking, dynamic time warping.

Résumé en arabe

تهدف الدراسة الحالية من خلال استخدام قواعد البيانات الطبية، إلى مساعدة الأطباء الجراحين حين تأدية عملهم. هذه القواعد متكونة من معلومات ومقاطع فيديو و وثائق تصف العمليات الجراحية. حين تأدية عملهم، من الصعب على الأطباء الجراحين البحث عن الوثيقة المماثلة أو الفيديو المماثل للوثيقة أو الفيديو المراد الاستعلام عنهما. لذالك، هدفنا تطوير أساليب و نظم تمكننا من البحث عن هذه الوثائق، توجيه إنذار للأطباء في حال توافق العملية مع أخرى قد واجهت مشاكل، أو رغبتهم في الاستعلام عن مرحلة من مراحل العملية الجراحية، أو معلومات عن المريض...نعتمد في ذالك على أساليب وطرق تستند على الاستنتاج وفقا للحالة (CBR) و البحث عن الفيديو بالمحتوى (CBVR) . يتم تقييم الطرق المطروحة اعتمادا على ثلاثة قواعد بيانات. قاعدتي بيانات لنزع غشاء العين (Pelage de membrane et la cataracte) تم إنشائها من قبل مختبرنا (LATIM) و أخرى تفيد في دراسة إمكانيات استخدام الأساليب المستعملة في مجال آخر(Hollywood).

لوصف مقاطع الفيديو نقترح ثلاث طرق. في الطريقة الأولى نعتمد علي تمثيل التحليل الحركي للفيديو عن طريق أعمدة و استخراج اتجاهه. تعتمد الطريقة الثانية على تقسيم بعض صور الفيديو إلى مناطق متناسقة الحركة و اتباعها من أجل استخراج مسارها. من خلال الطريقة الثانية، تبين لنا ترك معلومات تمكننا من وصف الفيديو بطريقة أكثر فعالية. لذالك قمنا بطرح طريقة ثالثة تعتمد على تلخيص الفيديو إلى عدة صور و من ثم استخراج مسارات المناطق. من أهم خاصيات الطرق المقترحة استعمالها لمجال الفيديو المضغوط (MPEG) الشيء الذي مكننا من استخراج المعلومات دون اللجوء إلى طرق أخرى في الغالب تكون أبطء.

بعد وصف الفيديو، تتم عملية البحث بحساب المسافة بين توصيف الفيديو المستعلم عنها و توصيفات جميع فيديو قاعدة البيانات، وذالك اعتمادا على مسافة مترية معينة. النتائج المتوصل إليها مشجعة.

كلمات مفاتيح : فهرسة، البحث عن الفيديو بالمحتوى، إتباع المناطق.

Remerciements

L'ENSEMBLE de ce travail a été effectué au Laboratoire de Traitement de l'Information Médicale, dirigé par le professeur Christian Roux.

En premier lieu, je tiens à exprimer ma reconnaissance et ma gratitude à Christian Roux, mon directeur de thèse, et à Guy Cazuguel, mon encadrant, pour m'avoir guidé avec justesse dans mon travail de recherche durant ces années. Merci pour leur confiance, leur aide et leur patience.

Je remercie tout particulièrement Mathieu Lamard, ingénieur de recherche au LaTIM, qui a été mon interlocuteur privilégié durant ce travail, et m'a apporté une aide présente en filigrane partout dans mon travail.

Je remercie Gwénolé Quellec, chargé de recherche à l'INSERM, pour son aide et ses conseils, qui m'ont été essentiels durant ces années de thèse.

Je remercie également M. Philippe POIGNET, professeur à l'université de Montpellier II, et Mme. Marie-Odile BERGER, directrice de l'équipe MAGRIT au Laboratoire LORIA (UMR 7503), INRIA Nancy Grand Est, pour avoir accepté la lourde tâche d'être mes rapporteurs.

Mes remerciements vont aussi à toute l'équipe du LaTIM et du département ITI : permanents, thésards, stagiaires, et chercheurs, qui m'ont accompagné au cours de mes années de thèse.

Un grand merci à mes parents, mon frère et mes deux soeurs, pour leur soutien durant toutes ces années d'études : je ne saurais être qu'infiniment reconnaissant quant aux sacrifices qu'ils ont consentis.

Et enfin, je tiens à remercier tous mes amis pour leurs encouragements et leur assistance, je leur adresse mes sincères amitiés.

Table des matières

Liste des figures

Liste des tableaux

Introduction

DURANT ces dernières années, la vidéo numérique a transformé le monde du multimédia avec de nouveaux supports de capture, de visualisation et de transmission des vidéos. La diffusion rapide et massive des avancées technologiques a bouleversé le paysage médiatique en multipliant l'offre grâce à des dispositifs d'acquisition accessibles à tous (caméras numériques, smartphones, ...), des écrans plats de moins en moins onéreux, qui sont vite devenus "HD", des capacités de stockage croissantes pour des coûts décroissants (disques durs, mémoires flash) et des systèmes de transmission et de diffusion via les réseaux Internet à haut débit spécialisés ou non. Dans ce contexte, il est devenu nécessaire de mettre à disposition des outils permettant de retrouver rapidement l'information désirée. Ce besoin a fait le succès des moteurs de recherche sur internet (Google, Yahoo, Bing, etc...). Leur caractéristique commune est qu'ils travaillent avec des éléments textuels, associés aux informations archivées, voire parfois avec des valeurs numériques. Le problème se pose quand on a affaire à des vidéos ou des images, qui nécessitent alors que des annotations textuelles soient associées à ces objets par des opérateurs. Ce sont des opérations coûteuses en temps, qui sont aussi sujettes à l'interprétation des annotateurs. Pour s'affranchir de ces problèmes, depuis une dizaine d'années, les chercheurs s'intéressent aux méthodes de recherche par le contenu, qui permettent de retrouver ce type d'objets en utilisant uniquement leur contenu numérique. C'est un domaine de recherche très dynamique [1–3] en particulier dans le domaine médical.

Les informations recueillies au cours d'un examen médical (signaux physiologiques, images, vidéos chirurgicales ou d'examens, analyses de sang, contexte clinique du patient, diagnostic du médecin, etc.) sont en effet de plus en plus souvent regroupées dans des dossiers patients spécialisés. L'intérêt de pouvoir recourir à des techniques de recherche par le contenu, à la fois textuel et numérique (images, vidéos...) dans des bases de données existantes est donc évident pour les praticiens. Cela leur permettrait de faire des recherches et des classements d'une manière plus efficace qu'actuellement. La création de ces bases de dossiers patients est un atout majeur pour le développement de nouvelles méthodes d'aide à la pratique médicale. Les dossiers stockés dans ces bases de données servent d'abord au suivi des patients, ils constituent une trace de leurs examens passés ; mais ils peuvent également être intéressants dans l'aide au diagnostic pour les nouveaux patients. Mais on peut envisager d'autres applications, parmi lesquelles l'aide en cours d'examen (guidage d'instruments en endoscopie) et l'aide au geste opératoire lors d'interventions chirurgicales. C'est dans ce contexte que se place ce travail de thèse.

Notre contribution

Dans ce travail de thèse, nous nous intéressons à l'utilisation peropératoire de bases de données pour l'aide à la décision, bases constituées à partir de l'information médicale multimédia acquise lors d'interventions chirurgicales antérieures. Le domaine cible est l'ophtalmologie et plus particulièrement les chirurgies de la rétine. Au cours de l'opération, le chirurgien est totalement concentré sur l'acte, et il ne peut donc pas aller consulter des dossiers et vidéos déjà archivés ; par contre des outils d'analyse automatique en temps réel

des images acquises en cours d'opération pourraient permettre cette utilisation de séquences déjà archivées, avec comme applications directes : des alertes en cas de problème, des informations sur les suites de tel ou tel geste dans des situations opératoires voisines (opération, caractéristiques patient, etc ...), des conseils sur les décisions. Nous nous plaçons donc dans le champ de la recherche de document multimédia par leur contenu numérique (CBVR : Content Based Video Retrieval). Nous présentons différentes méthodes pour pouvoir traiter des requêtes en cours d'opération, la requête étant alors la vidéo du champ opératoire. Ces méthodes sélectionnent les dossiers et vidéos les plus proches de la requête au sein d'une base de données. On peut alors se baser sur les scénarios et les annotations effectués par d'autres experts pour proposer de l'aide aux chirurgiens. Si la recherche de documents multimédia par le contenu est un sujet de recherche particulièrement actif depuis une dizaine d'années [4], la recherche de dossiers médicaux contenant notamment des vidéos, est un sujet extrêmement novateur et récent, et peu de publications abordent le sujet [5].

Cela soulève cependant de nouvelles questions. Tout d'abord, comment caractériser des vidéos de chirurgies : peut-on se baser sur des méthodes utilisées pour des vidéos de télédiffusion par exemple, qui ont des caractéristique différentes de celles des vidéos médicales ? Comment comparer ces vidéos sachant que des chirurgies de même type n'ont pas forcément les mêmes déroulements, dans la suite des gestes opératoires, mais aussi dans leur durée, qui dépend de l'expertise du médecin ? Comment également introduire un système de recherche de vidéos dans le processus chirurgical, en temps réel ?

Dans notre travail de thèse, nous apportons quelques réponses et quelques pistes à ces questions. Pour rechercher les vidéos les plus proches d'une vidéo placée en requête, nous nous intéressons à la recherche de vidéo par leur contenu numérique (Content-Based Video Retrieval, CBVR). Pour ce faire, nous devons définir une mesure de similitude entre deux vidéos. Elle s'appuie sur la définition de signatures associées à chaque vidéo. Une signature est constituée d'informations caractérisant la vidéo. Pour construire ces signatures, nous nous basons sur le contenu numérique des vidéos. Les vidéos déjà enregistrées pourraient en effet être décrites, caractérisées par des médecins mais outre le temps que cela demanderait, ce ne serait pas possible en cours d'intervention. Nous extrayons donc, dans les données numériques, des éléments permettant de les caractériser. Les contraintes de calcul en temps réel nous ont incités à nous intéresser à des méthodes pouvant utiliser directement les données vidéo générées par les systèmes de compression vidéo. Ces données, pour des taux de compression raisonnables (pas ou peu de perte d'information visuelle), contiennent toute l'information nécessaire. Nous avons développé des méthodes de construction de signatures basées sur ces données de compression, qui nous permettent d'accéder à des éléments caractérisant les vidéos d'une manière rapide et efficace sans faire de décomposition totale de la vidéo.

La suite de ce manuscrit est organisée de la façon suivante :

- Le chapitre 1 rappelle les approches principales de la recherche d'information par le contenu et du raisonnement à base de cas. Nous commençons par une description des approches des systèmes CBIR et CBVR, leurs éléments et l'architecture de base de tels systèmes. Nous donnons ensuite un état de l'art des systèmes CBVR avec différentes approches. Ceci permettra de mieux comprendre le domaine de recherche ciblé, et le cadre de cette thèse.
- Le chapitre 2 expose la norme de compression de vidéo utilisée pour l'extraction des paramètres. Nous présentons une description globale du processus de compression de vidéo, puis, le principe de fonctionnement des codeurs vidéo, et plus particulièrement le codeur H.264/AVC, celui sur lequel nous nous appuierons pour créer des signatures de vidéo pour la CBVR.

- Le chapitre 3 présente les méthodes d'indexation développées pour la CBVR, en se basant sur les paramètres extraits depuis la norme H264/AVC. Nous proposons trois méthodes différentes pour la génération des signatures des vidéos, ainsi que, les mesures de similitude associées.
- Le chapitre 4 est consacré à la description des bases de vidéos utilisées et à la présentation et discussion des résultats obtenus. Les résultats sont comparés aux résultats publiés dans des travaux antérieurs.
- Dans la conclusion, nous essayerons de dégager des perspectives.

CHAPITRE 1

INDEXATION ET RECHERCHE D'INFORMATION PAR LE CONTENU

Du fait des avancées dans le domaine des technologies de l'information et de la Communication (télévision numérique, Internet, etc.), le volume de données numériques échangées et archivées s'est accru de façon spectaculaire. De ce fait, l'accès automatique et rapide à l'information pertinente devient une tâche fondamentale mais complexe. Elle constitue un domaine de recherche en pleine expansion. Ce chapitre a pour but de rappeler les concepts généraux de ce domaine du traitement de l'information. Nous y présentons les approches principales de la recherche d'information, de son indexation et les problèmes associés.

1.1 Recherche d'information par le contenu

La Recherche d'Informations (RI) consiste à sélectionner dans une collection de documents ceux susceptibles d'être pertinents vis à vis d'un besoin en information d'un utilisateur. Ce besoin en information est généralement exprimé par une requête. La qualité d'un système de recherche d'informations vis à vis d'une requête est évaluée par rapport à plusieurs critères. Le premier concerne la façon d'exprimer une requête. Le second critère concerne les fonctions de la recherche d'informations, qui vont permettre de restituer les documents pertinents par rapport à la requête. Pour évaluer l'adéquation entre un ensemble de documents et une requête, les systèmes de recherche d'information doivent posséder d'une part une représentation interne de l'information dans les documents disponibles et dans la requête utilisateur, et d'autre part d'une méthode de comparaison afin de déterminer leur degré de correspondance. Les représentations internes ainsi que la manière de les comparer définissent le modèle de recherche.

Les systèmes les plus connus et les plus anciens sont les recherches par mots-clefs dans des documents textes. Systèmes qui ont vu comme généralisation la recherche en texte intégral que nous proposent aujourd'hui tous les moteurs de recherche sur internet. Dans notre travail, nous nous intéressons à des systèmes qui travaillent non pas sur des documents textes seuls, mais sur des documents multimedia, contenant en particulier des images, des vidéos, qui ne sont pas décrites par des textes. Il faut donc trouver des méthodes pour les comparer qui s'appuient sur leur contenu numérique (valeurs des pixels, relations entre eux, formes, objets, etc..) : c'est la recherche d'information par le contenu. Les principes de base restent les mêmes,

mais il faut trouver des représentations de l'information contenue dans ces données.

Plusieurs éléments ressortent de cette définition et constituent le système de recherche d'information par le contenu :

- une représentation des documents ou signature du document, qui sert comme caractéristique pour reconnaître le document et le distinguer parmi les autres. Elle est souvent considérée comme un index du document, de la même manière qu'on définit des index dans les livres ou les systèmes d'archivage. D'ou le terme "Indexatio" associé très souvent à la recherche d'images/vidéos par le contenu.
- une métrique de similitude (ou de distance) qui permet de comparer les signatures.
- des algorithmes de recherche qui, basés sur les deux outils précédents, permettent de retrouver rapidement les documents recherchés.
- une interface utilisateur, qui rend transparente la procédure de recherche et facilite l'introduction de la requête.

Les experts humains sont a priori les mieux placés pour construire la signature des documents, en les décrivant par exemple par rapport à une nomenclature, une représentation des informations contenues dans des documents particuliers d'un domaine. Dans le domaine médical, par exemple, il existe des descriptions normalisées des organes, des lésions, etc. Cependant, cette opération est très coûteuse en temps et difficilement réalisable, étant donnée la taille énorme des bases d'images. D'où l'intérêt de l'indexation automatique.

1.2 Le raisonnement à base de cas

1.2.1 Cadre général

Le cadre général de notre travail est l'aide à la décision dans le domaine médical, plus particulièrement pour la prise de décisions chirurgicales. Les informations recueillies au cours d'un examen (images, vidéos chirurgicales (opérations), analyses de sang, contexte clinique du patient, diagnostic du médecin, etc.) sont de plus en plus souvent regroupées dans des dossiers patients spécialisés. La création de ces bases de dossiers patients est un atout majeur pour le développement de nouvelles méthodes d'aide à la pratique médicale. De tels systèmes peuvent profiter utilement des informations contenues dans des cas cliniques déjà analysés et enregistrés, en y retrouvant par exemple des vidéos du même type que l'opération prévue ou des situations opératoires voisines (caractéristiques patient, etc). Ils permettront aux médecins d'utiliser l'expérience d'autres praticiens, de comparer les pratiques et donc enrichir leurs connaissances, mieux réagir peut-être dans des situations qu'ils n'ont pas rencontrées. La constitution de bases de dossiers patients, l'étude de méthodes permettant de les comparer est la base de l'aide à la décision par raisonnement à base de cas. Cette méthode se formalise pour pouvoir être mise en oeuvre en informatique, dans des systèmes de recherche à base de cas [1].

1.2.2 Principes généraux de raisonnement à base de cas

Le raisonnement à base de cas (CBR-Case Base reasonning) ou raisonnement à partir de cas, est une approche de résolution de problèmes qui utilise des expériences passées pour résoudre de nouveaux problèmes [2]. L'ensemble des expériences forme une base de cas. Typiquement un cas contient au moins deux parties : une description de situation représentant un "problème" et une "solution" utilisée pour remédier à cette situation. Parfois, le cas décrit

également les conséquences résultant de l'application de la solution (succès ou échec). Les techniques CBR permettent de produire de nouvelles solutions en extrapolant sur les situations similaires au problème à résoudre. Cette approche est adéquate pour les domaines où la similarité entre les descriptions de problèmes nous donne une indication de l'utilité des solutions antécédentes.

Au début de la dernière décennie, on a assisté à un regain de popularité du domaine et de nouvelles tendances qui misent sur la simplification de la représentation des cas et sur des applications à grande échelle. Les travaux initiaux sur le sujet remontent aux expériences de Schank et Abelson en 1977 [3]. Le raisonnement à base de cas connaît un développement croissant dans le domaine médical [4]. Il est néanmoins encore assez méconnu par rapport à d'autres technologies du domaine des sciences cognitives, comme la fouille de données (data mining) qui est l'exploration et l'analyse de grandes quantités de données afin d'y découvrir de l'information implicite. Les objectifs ne sont non plus pas identiques. Avec le raisonnement à base de cas, on cherche des solutions à des problèmes précis, par exemple une aide au diagnostic.

L'approche CBR offre de nombreux avantages. Pour certaines applications, l'approche CBR est plus simple à mettre en oeuvre que les approches basées sur un modèle du domaine (base de règles) ; elle permet d'éviter les problèmes d'acquisition de connaissance ("knowledge bottleneck") qui rendent difficile la conception de bases de connaissances de taille importante. Le CBR est particulièrement bien adapté pour les applications dont la tâche est accomplie par des humains expérimentés dans leur domaine et dont les expériences sont disponibles dans une base de données, ou des documents. On l'utilise pour les domaines n'exigeant pas de solution optimale et dont les principes sont mal formalisés ou peu éprouvés.

Nous nous appuyons sur le modèle générique (voir figure 1.1) pour présenter les étapes de la CBR :

Représentation : la représentation des cas est très importante dans la réalisation d'un système CBR. En effet, cette représentation va déterminer l'efficacité et la rapidité de la recherche des cas dans la base. Il est donc nécessaire de choisir les informations à stocker dans chaque cas sous la forme adéquate. Un cas est décrit par de nombreuses caractéristiques représentant différents types d'informations. Généralement on considère les cas comme une liste de couples attribut-valeur. Chaque couple correspondant à une caractéristique.

Recherche : cette phase permet de déterminer les cas de la base qui sont les plus similaires au problème à résoudre. Plusieurs heuristiques, telles que l'algorithme des plus proches voisins (Nearest Neighbour) [5], peuvent être utilisées pour mesurer la similitude entre la requête et les cas de la base.

Réutilisation : on propose pour le cas placé en requête, les solutions associées aux cas de la base les plus proches.

Révision : après sa génération par la méthode, la solution du problème est testée. Si l'aide est correcte, le cas est retenu : c'est la phase de conservation. Si la solution n'est pas satisfaisante, il faut la corriger : c'est la phase de révision.

En médecine, le système CBR doit pouvoir intégrer de l'information symbolique, telles que des annotations cliniques et des informations numériques comme des images, des vidéos, etc. La CBR permettra de raisonner par similitude, à la fois pour indexer et rechercher des dossiers patient (examens par vidéos, compte-rendu examens, etc.). Il doit aussi combiner des techniques d'indexation et de recherche fondées sur des critères visuels numériques avec des techniques fondées sur des critères symboliques. Certaines méthodes ont été mises au point

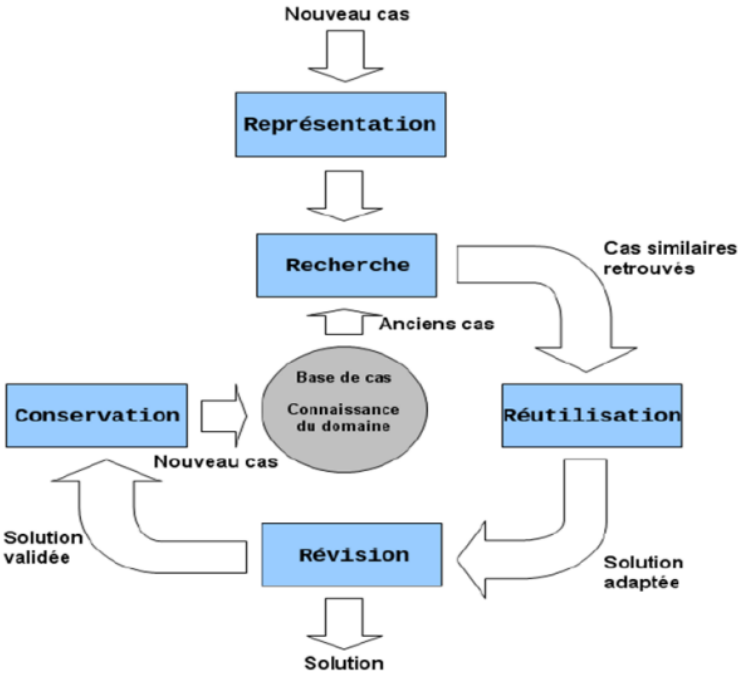


Figure 1.1 — Les étapes du raisonnement à base de cas

pour gérer de l'information symbolique [6]. D'autres méthodes, basées sur la recherche par le contenu, ont été mises au point pour gérer des images, des vidéos [7]. Dans la suite de ce chapitre, nous commençons par une présentation des systèmes de recherche d'informations basés sur le contenu avant de poursuivre par une présentation plus détaillée de la recherche de vidéo par le contenu (CBVR), système auquel nous nous intéressons. Nous parcourons les concepts et les approches basées sur l'extraction de caractéristiques numériques de la vidéo. Nous expliquons ensuite la tendance de la recherche dans ce domaine et les principaux axes que nous développons dans les chapitres qui suivent.

1.3 Architecture des systèmes d'indexation des bases d'images et de vidéo

Les systèmes de recherche d'informations basés sur le contenu, font intervenir deux phases qui sont l'indexation et la recherche.

1.3.1 Indexation

L'indexation consiste à extraire, représenter et organiser efficacement l'information contenue dans des documents d'une base de données (figure 1.2). Pour cela, les documents sont tout d'abord représentés par une signature numérique qui permettra leur identification, et leur comparaison. Cette opération est réalisée en deux étapes. La première étape implique l'analyse des documents pour en extraire les caractéristiques pertinentes au vu de l'usage visé. Par exemple, si on veut comparer des images de rétinopathies diabétiques, par rapport au nombre de microanévrismes présents dans la rétine, il faudra détecter ces lésions et les compter. La seconde étape permet éventuellement de compresser l'information extraite sans

nuire à l'efficacité de représentation de la signature. Il est important d'avoir des signatures compactes pour éviter d'avoir des données trop importantes à stocker et à traiter. Les signatures sont organisées au mieux afin d'optimiser la recherche de l'information. Dans certains cas, une structure hiérarchique est utilisée pour organiser et faciliter l'accès aux signatures. La communauté scientifique du domaine a cherché très vite à inclure automatiquement du contenu sémantique dans les signatures construites initialement avec uniquement le contenu visuel [8].

1.3.1.1 Description numérique

Des descripteurs de bas niveau, tels que la couleur, la texture, la forme, et le mouvement, peuvent être associés aux objets ou des régions dans l'image. La couleur des images choisie peut être décrite par l'histogramme de couleur ou par les couleurs dominantes. Les paramètres de mouvement de caméra et le taux d'activité décrivent le mouvement au niveau du la vidéo. Le mouvement des objets peut être décrit par des trajectoires. Les sommaires d'images clés sont généralement employés. Les images clés, qui se rapportent à une ou plusieurs images représentatives dans un plan, fournissent une représentation compacte.

Nous reprendrons en détail les différents descripteurs dans la section (§1.6.4).

1.3.1.2 Description sémantique

L'information sémantique peut être représentée par des annotations structurées ou du texte libre (mots clés), ou par des modèles sémantiques. Les annotations peuvent être manuelles, ou extraites automatiquement à partir du sous-titrage, par la détection et l'identification de visages, de décors, ou d'actions modélisés spécifiquement. Les modèles sémantiques peuvent décrire des entités, telles que des objets et des événements, et des relations entre elles, qui rendent possible le traitement de requêtes complexes. Certains modèles sémantiques sont considérés comme des prolongements des modèles d'Entité Relation (ER) développés pour des documents par les communautés de recherche dans les bases de données et les bases documentaires.

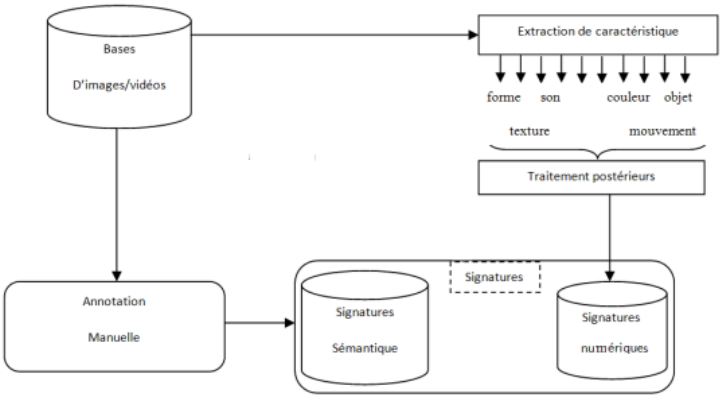


Figure 1.2 — Phase d'indexation par le contenu

1.3.2 Recherche

La recherche d'informations est l'ensemble des opérations nécessaires pour répondre à la demande d'un utilisateur (figure 1.3). Tout d'abord, l'utilisateur doit construire une requête. Cette opération, évidente pour le texte, est bien plus difficile pour les images et encore plus pour la vidéo. La requête peut inclure différentes données : un exemple (image, vidéo, son), un dessin ou une animation. En règle générale, la difficulté est d'exprimer correctement l'objet de la requête en utilisant au mieux les moyens proposés par le système. La requête est ensuite transformée en signature en suivant un procédé similaire à l'indexation. Cette signature est alors comparée aux signatures de la base de données afin de retrouver l'information la plus pertinente. Toutefois il est particulièrement difficile de répondre aux exigences des utilisateurs à partir d'une seule requête. Il est alors utile d'intégrer un bouclage de pertinence incluant l'avis de l'utilisateur pour améliorer la requête en fonction du résultat précédemment obtenu. Un tel système permet également à l'utilisateur de clarifier sa demande qui est souvent mal formulée au début de la recherche.

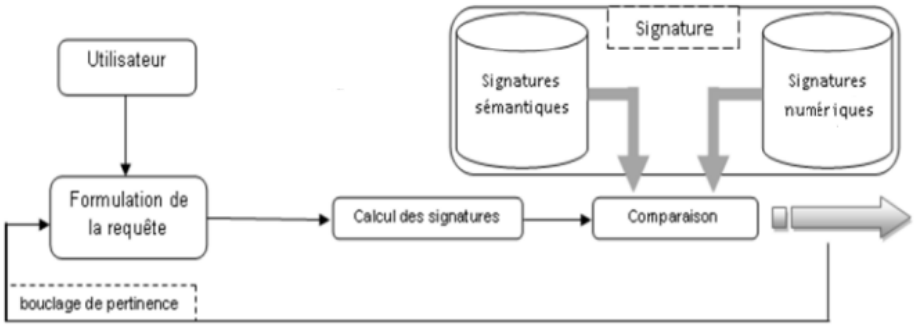


Figure 1.3 — Phase de recherche par le contenu

1.4 Critères usuels de performance d'une méthode de recherche d'informations

L'évaluation des algorithmes de recherche est une tâche très complexe. Elle doit prendre en compte les performances qualitatives des résultats fournis à l'utilisateur, mais aussi le temps de recherche ou la taille de la signature par exemple. L'évaluation des méthodes s'appuie sur deux étapes principales. Il faut d'abord définir avec précision le critère d'évaluation, puis la mesure d'évaluation associée à ce critère.

La qualité d'une méthode de recherche d'informations peut être jugée par un grand nombre de critères différents. Ces critères peuvent être groupés en plusieurs classes :

- l'effectivité : la pertinence, la capacité de discrimination, la stabilité par rapport à des changements de la requête, l'intégrité des résultats, la complexité de formulation de la requête, etc.
- l'efficience : le temps de recherche, le temps pour donner le résultat de la recherche, le temps pour la génération des index, le temps d'insertion, l'espace de stockage des index, le temps pour la génération d'une requête, etc.
- la flexibilité : l'adaptabilité, capacité à généraliser, etc.
- autres : la présentation des résultats, etc.

Chaque classe possède plusieurs sous-critères et chacun de ces sous-critères doit être évalué individuellement pour obtenir une évaluation globale de la méthode.

La deuxième étape dans le processus de l'évaluation est de définir les mesures associées aux critères d'évaluation. Elles sont simples pour certains critères (comme le temps de recherche). Mais ce n'est malheureusement pas aussi simple pour la majorité des critères cités. Le critère auquel nous allons nous intéresser principalement est la capacité de discrimination (que nous appelons aussi efficacité de retrouvaille). L'objectif d'une méthode de recherche est de retrouver les documents les plus proches de la requête, pour une mesure de similitude donnée. L'efficacité globale de la méthode peut être mesurée uniquement si les similitudes réelles sont connues, ce qui suppose pour une méthode automatique une classification des documents. En général, une évaluation des méthodes de recherche demande :

1. une collection de N documents (la base de données).
2. un ensemble de M requêtes de référence.
3. un ensemble de métriques d'évaluation.

La pratique commune pour évaluer l'efficacité de retrouvaille (retrieval en anglais) est la suivante : une requête est présentée au système, le système renvoie une liste de k documents classés en fonction de leur degré de similitude avec la requête, fonction de la métrique utilisée ; puis, pour chaque valeur de k (= nombre de documents présentés en réponse à la requête, que nous appellerons "fenêtre de retrouvaille"), les valeurs suivantes sont calculées (V_n est la pertinence du document n, $V_n = 1$ si la requête et le document n présenté en réponse appartiennent à la même classe, $V_n = 0$ sinon) :

– les détections (équation 1.1) : le nombre d'objets appropriés extraits

$$A_k = \sum_{n=0}^{k-1} V_n \tag{1.1}$$

– les faux positifs (équation 1.2) : documents retrouvés par la recherche mais ne correspondant pas à la requête

$$B_k = \sum_{n=0}^{k-1} (1 - V_n) \tag{1.2}$$

– les faux négatifs (équation 1.3) : documents appropriés à la requête mais non retrouvés par la recherche

$$C_k = \sum_{n=0}^{N-1} V_n - A_k \tag{1.3}$$

Les mesures de performance usuelles du domaine de la recherche d'information sont ensuite calculées :

– le rappel (équation 1.4) : rapport entre le nombre d'objets appropriés extraits et le nombre d'objets appropriés (extraits et non extraits) dans la base de données.

$$R_k = \frac{A_k}{A_k + C_k} \tag{1.4}$$

– la précision (équation 1.5) : rapport entre le nombre d'objets appropriés extraits et le nombre total d'objets extraits (appropriés et non appropriés).

$$P_k = \frac{A_k}{A_k + B_k} \tag{1.5}$$

La précision et le rappel donnent une bonne indication de la performance de la méthode (ils prennent des valeurs entre 0 et 1 ; les valeurs élevées, voisines de 1, indiquent une bonne performance). Mais une mesure seule est insuffisante. Nous pouvons toujours avoir le rappel égal à 1, simplement en donnant à k une valeur égale à la taille de la base. De même, la précision gardera des valeurs élevées en recherchant seulement quelques documents ($k \ll$ taille de la base). Ainsi, la précision et le rappel sont en général utilisés ensemble (par exemple, la valeur de précision où le rappel est égal à 0.5), ou le nombre de documents proposés (valeur d'arrêt) est indiqué (par exemple, le rappel quand 100 images sont affichées, ou la précision pour 20 images). Le nombre k de documents proposés est choisi par l'utilisateur. Dans la pratique, ce nombre est choisi pour que ces k documents soient visualisés commodément. Cependant, les mesures sont sensibles au choix du nombre k. Si le nombre est petit, les petites différences dans l'exécution des algorithmes peuvent mener à de grandes différences dans la précision et le rappel. D'autre part, de grandes valeurs de k ne permettent pas de distinguer les différences de performance. Par conséquent, les deux mesures sont souvent calculées pour différentes valeurs de k et représentées sur le même graphique : nous obtenons une courbe paramétrée par k. Le résultat graphique est appelé graphique "précision - rappel" comme représenté sur la figure 1.4.

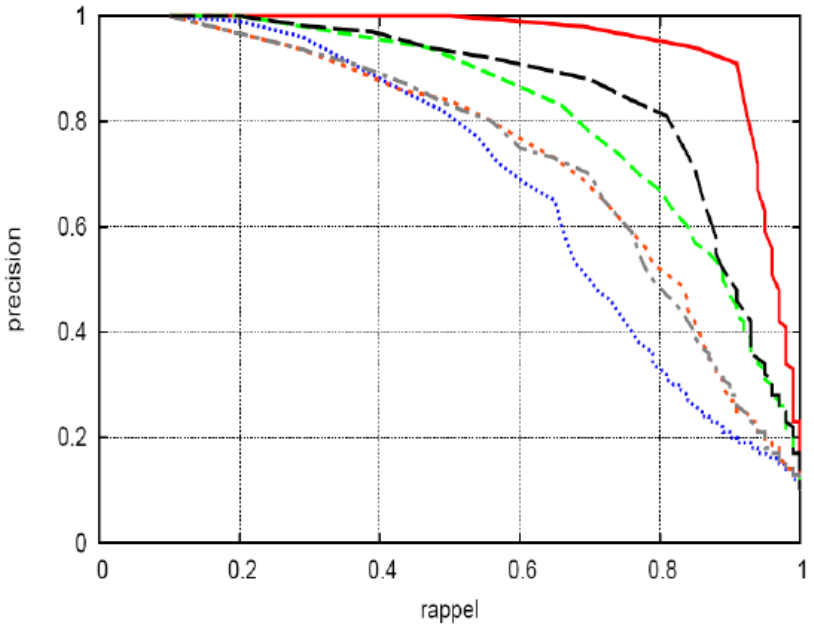


Figure 1.4 — Courbes de précision-rappel. Plusieurs courbes de précision-rappel sont présentées sur la figure, chacune étant associée à une méthode de recherche. La méthode la plus performante est celle dont la courbe est le plus à droite : les valeurs de précision sont les plus élevées pour toutes les valeurs de rappel.

1.5 La recherche d'image par le contenu (CBIR)

Depuis le début des années 90 l'indexation et la recherche d'image par le contenu sont devenues un domaine très actif de la recherche et de nombreux systèmes commerciaux et académiques ont été proposés [9]. Puis rapidement des extensions sont apparues pour réaliser des systèmes d'indexation et de recherche de vidéos. Les premiers systèmes introduits concernaient l'indexation et la recherche d'images. l'utilisation du terme « recherche d'images par

le contenu ≫ dans la littérature a été faite par T. Kato [10], pour décrire ses expériences sur la recherche automatique d'images dans une base de données en utilisant des caractéristiques de bas niveau comme la couleur et la forme. A partir de là, le terme a été utilisé pour décrire le processus de recherche d'images au sein d'une grande collection d'images d'après des caractéristiques qui peuvent être extraites automatiquement des images elles-mêmes (telles que la couleur, la texture et la forme). Les caractéristiques utilisées pour la recherche peuvent être quantitatives (numériques) ou sémantiques, mais le processus d'extraction doit être de préférence complètement automatique. Sans aucun doute, la recherche d'images par les mots-clés assignés manuellement n'est pas de la CBIR telle qu'on l'entend généralement même si les mots-clés décrivent partiellement le contenu de l'image. Pour caractériser/indexer les images, la CBIR emploie beaucoup de méthodes utilisées par le traitement d'images et la vision par ordinateur, et est considérée par certains comme un sous-ensemble de ce champ. Le traitement d'images couvre un champ beaucoup plus large, y compris l'amélioration d'images, la compression, la transmission, l'interprétation. Lorsqu'on travaille dans des bases d'images spécialisées (par exemple dans des bases médicales spécifiques), la CBIR permet de construire des signatures d'images plus spécifiques, contenant des informations très ciblées. Par conséquent, la précision des méthodes d'indexation proposées peut donc être considérablement améliorée par rapport au cas général Les méthodes de recherche d'images par le contenu, spécifiques au domaine médical, peuvent être regroupées en plusieurs catégories :

- les méthodes basées sur la segmentation, qui permettent d'extraire des formes d'intérêt telles que des régions, des objets caractéristiques, etc... [11, 12]. En général, il est difficile d'extraire automatiquement toutes les formes d'intérêt. Ainsi, des experts médicaux sont sollicités pour déterminer des régions d'intérêt (human/physician in the loop approach) [13].
- les méthodes utilisant directement la description des régions d'intérêt faite manuellement par les médecins [14]
- les méthodes consistant à caractériser l'agencement des formes intéressantes (organes, lésions, ...) présentes dans l'image à l'aide d'un graphe topologique, qui sert alors d'index à l'image [15, 16]
- les méthodes basées sur l'extraction de descripteurs bas niveau connus pour bien caractériser les pathologies étudiées [16].

Il faut ensuite définir des métriques de simililitudes pour chaque méthode, pour comparer les images.

1.6 La recherche de vidéo par le contenu (CBVR) : etat de l'art

Les vidéos contiennent deux sources supplémentaires d'information par rapport à l'image. Il s'agit du mouvement et du son. La recherche de vidéo par le contenu (CBVR) dépasse donc largement, par sa complexité, le cadre habituel des techniques d'indexation dédiées aux données monomédias (audio, image fixe, ...). Il sagit ici d'indexer des contenus dynamiques, riches et hétérogènes, faisant intervenir différents types de média (image, son, texte, ...) et un très gros volume d'information.

Plusieurs groupes de chercheurs ont étudié les possibilités d'adapter les techniques de la CBIR à la recherche de vidéos par le contenu. Cette adaptation permet d'avoir comme élément d'indexation des descriptions de couleur, de forme, de position spatiale des objets visuels, et trajectoire de mouvement spécifique. Tous ces descripteurs de bas niveau peu-

vent former une base pour des signatures (index). L'indexation et la recherche de vidéos (CBVR) trouvent évidemment leur place dans des applications variées, au premier chef pour la gestion des vidéos multimédias de type télédiffusion..., elles sont aussi utilisées dans le domaine de la vidéo surveillance [17], où on peut distinguer deux catégories d'application. La première catégorie est la sécurité, une alarme est déclenchée si le système détecte un événement anormal. Habituellement, le personnel de sécurité veut trouver des informations antérieures concernant le ou les objets impliqués dans cet événement. La deuxième catégorie est l'étude statistique. Il est intéressant de savoir combien de fois par mois un événement aura lieu ou quel événement suit un événement particulier. De nombreux projets de recherche ont été menés à travers le monde. L'agence américaine pour les projets de recherche avancée de la défense, DARPA (Defense Advanced Research Projects Agency), par exemple, a fondé un projet multi-institutionnel sur la vidéo surveillance et le contrôle, VSAM (Video Surveillance And Monitoring) [18].

D'autres applications sont plus orientées vers l'indexation des gestes réalisés [19, 21], où l'indexation et la recherche de vidéos de sport basées sur le contenu est une application possible. Dans un contexte de football par exemple, l'utilisateur pourrait interroger une base de données de matches pour obtenir tous les passages d'un joueur par exemple, sans qu'il y ait eu au préalable une indexation manuelle. Cela permettrait d'éviter à un opérateur humain de parcourir l'ensemble d'une base de données non indexée. Plus récemment, la CBVR a été introduite dans d'autres domaines ; en particulier dans, le domaine médical pour l'interprétation clinique et la formulation d'un diagnostic. Dans ce contexte, on peut distinguer deux type d'applications : la recherche de vidéos médicales par le contenu et la chirurgie assistée par ordinateur.

Un système de recherche d'images et de vidéos médicales a été proposé par Peijiang Yuan et all, pour fournir aux médecins une aide au diagnostic, en se basant sur des informations médicales multimédias acquises auprès de plusieurs patients, et déjà archivées dans des bases de données [22]. Un travail similaire a été proposé concernant plus des données vidéo issues de la chirurgie de la rétine [23]. Une autre application introduit l'Endomicroscopie Confocale par Minisondes (ECM) [24], celui-ci permet l'observation dynamique des tissus au niveau cellulaire pendant une endoscopie. Le but principal de l'étude était d'assister les endoscopistes dans l'interprétation in vivo des séquences d'images ECM, en mettant en disposition un système de reconnaissance de vidéos endomicroscopiques capable d'extraire automatiquement dans une base de données, plusieurs vidéos ECM qui ont une apparence similaire à la vidéo requête, mais qui ont déjà été annotées avec différentes métadonnées telles que par exemple le diagnostic histologique. Zelnik-Manor et Irani proposent un système d'analyse de comportement chez les patients dans le but de les reconnaitre [25]. L'analyse permet d'étudier le comportement et rechercher les cas similaire dans une base de données afin de fournir une aid au diagnostic.

Peu de publications ont concerné la chirurgie assistée par ordinateur. Dans [26], un système de représentation de haut niveau de l'information visuelle est présenté ; il reflète non seulement la structure mais aussi la chirurgie de la sémantique sous-jacente et le contexte de l'environnement in vivo. L'objectif est de faciliter l'analyse du flux de travail chirurgical et sa compréhension. En particulier, une représentation basée sur les données issues de la chirurgie mini-invasive (MIS) a été proposée. Liao et all proposent une technique de super-position autostéréoscopique d'image 3D intégrée dans un système de navigation chirurgicale sur le patient en utilisant un miroir semi-argenté [27]. Ce système augmente la précision de la chirurgie et réduit les cas envahissant. Cao et all introduisent une nouvelle technique pour la détection d'instruments thérapeutiques et la détection des phases chirurgicales [28]. Des applications similaires sont proposées dans [29] et [30]. Une technique de localisation 3-D pour les instru-

ments chirurgicaux à partir séquences vidéo laparoscopie est proposée dans [31]. Le but est d'aider au développement d'applications de réalité augmentée en chirurgie. Des applications similaires ont été données dans [32] ; en raison de la limitation du champ de vue, qui peut causer des difficultés de navigation, l'outil combine un grand nombre d'endo-images vidéo microscopiques automatiquement. Le champ de mouvement entre des images successives est estimé par le flux optique. Ensuite, l'image mosaïque est construite.

Par rapport à tous ces travaux, l'objectif de notre travail est assez original. Nous voulons proposer des méthodes de caractérisation et de comparaison de vidéos pour l'aide à la prise de décision lors d'interventions chirurgicales, réalisées sous microscope et/ou contrôle vidéo (cadre de la chirurgie assistée par ordinateur). Elles utiliseront des vidéos déjà enregistrées et commentées. Notons que ces méthodes pourraient aussi être utilisées pour étudier des examens vidéos type examens endoscopiques. Avant de présenter au chapitre 3 les méthodes que nous avons développées, il nous faut rappeler quelques définitions sur les vidéos. Notre objectif étant de trouver des caractérisations des vidéos, nous verrons ensuite comment différencier les différentes phases d'une vidéo (segmentation), pour pouvoir les caractériser, avant d'examiner les différentes possibilités de description de ces phases.

1.7 Indexation et structuration des vidéos

Plusieurs techniques d'indexation ont été présentées dans la littérature. Les descripteurs numériques de la vidéo correspondent généralement à des interprétations en termes de couleur, de texture et de forme. Ces informations résultent de l'analyse de chaque image ou de segments d'images de la vidéo. Extraire les descripteurs de chacune des images d'un document vidéo rendrait le temps de traitement prohibitif. D'autre part, il faut prendre en compte le fait que deux images consécutives dans le même plan d'une vidéo sont assez semblables, et, si ce n'est pas le cas, cette différence est porteuse d'information au niveau structurel, puisqu'elle peut correspondre à un changement de plan. Cette observation conduit donc à favoriser un traitement de segments d'images, plutôt qu'un traitement image par image. De nombreuses approches réduisent ainsi le problème de l'extraction du contenu d'un segment d'images au traitement des descripteurs d'une seule image du segment considéré. Tous ces descripteurs peuvent former une base pour des signatures (index). Pour les mettre en oeuvre, il est nécessaire de disposer d'algorithmes de segmentation vidéo. La segmentation vidéo comprend la segmentation temporelle, telle que la détection des changements de plan et la détection d'effets de transition spéciaux, et la segmentation spatio-temporelle, telle que la segmentation en objets et leur suivi. La segmentation temporelle de la vidéo passe souvent par une segmentation en plans, scènes puis en images représentatives des plans afin de conserver uniquement l'essentiel du contenu.

1.7.1 Structure d'une vidéo

Une vidéo est un enregistrement dans le temps, sur un support physique, d'une scène d'origine optique ou provenant d'un appareil d'acquisition qui convertit l'information sous forme de séquences vidéos (exemple : échographes)). Une vidéo peut être représentée sous forme d'images synchronisées avec du son, dans un espace de temps discret. On les représente généralement à travers plusieurs niveaux hiérarchiques : le document complet, les séquences, les scènes, les plans, puis les images.

– **Séquence**

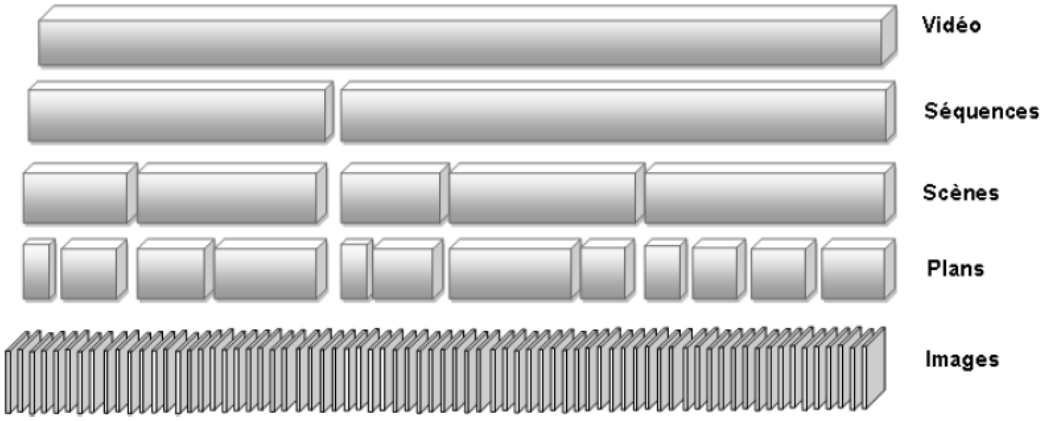


Figure 1.5 — Structure cinématographique d'une vidéo

Une séquence est un ensemble de scènes. Elle constitue une unité de sujet (par exemple un reportage dans un journal télévisé).

- **Scène**
  une scène est un ou plusieurs plans qui partagent le même contenu en terme d'actions, de lieux et de temps. Ces plans sont souvent consécutifs.
- **Plan de vidéo (video shot)**
  Un plan vidéo est une séquence d'images qui sont acquises de manière continue par une seule caméra. Le plan de vidéo est une unité de base de vidéo. Une vidéo peut être représentée par un ensemble de ses plans.
- **Images clés (keyframe)**
  Une image clé est une image qui représente significativement le contenu visuel d'un plan de vidéo. Selon la complexité du plan, une ou plusieurs images clés peuvent être choisies.

1.7.2 Méthodes de segmentation de vidéo

1.7.2.1 Segmentation en plans

La segmentation en plans est la technique de segmentation temporelle des enregistrements vidéo la plus répandue et la plus utilisée. Les méthodes de détection de changements de plan localisent les images, à travers lesquelles de grandes différences sont observées dans un certain espace de caractéristiques [33–35]. L'espace de caractéristiques se compose habituellement d'une combinaison de couleurs et de mouvements. Les changements de plan peuvent être instantanés ou apparaître sur plusieurs images, appelées les effets de transition progressifs, tels que les fondus et les volets. Il est plus facile de détecter des transitions instantanées que des effets progressifs.

La méthode la plus simple pour la détection du changement de plans est d'analyser les variations d'intensité des pixels entre les images successives. Si un nombre prédéterminé de pixels montre des différences plus grandes qu'une valeur seuil, alors l'occurrence d'un changement peut être déclarée. Une approche légèrement différente consiste à diviser chaque image en blocs rectangulaires, à opérer des évaluations statistiques dans chaque bloc indépendamment, et à vérifier alors que le nombre de blocs qui ont globalement été modifiés est supérieur à un seuil. Les deux approches peuvent être sensibles au bruit et à la compression. Cependant,

il existe de nombreuses solutions qui s'appliquent à la vidéo de manière générique avec une précision plus qu'acceptable [33–35].

La micro-segmentation est une segmentation temporelle à une échelle encore plus petite que celle du plan. Elle est basée sur la segmentation en événements, en mouvements de caméra, en entrée-sortie d'objets ou de personnages [36]. Par opposition, la macro segmentation effectue une segmentation qui se rapproche de la composition sémantique des documents (segmentation en séquences, en chapitres, en programmes) [37].

1.7.2.2 Segmentation en scène

La scène est une unité sémantique importante puisqu'elle contient une séquence de plans dont la logique permet d'exprimer une idée. Par contre, le plan est une unité technique qui est souvent de courte durée et son étude isolée ne permet pas de comprendre réellement le déroulement de la scène. La détermination des scènes est alors utile pour la navigation, la visualisation des données audiovisuelles et aussi pour leur analyse sémantique. Elles permettent aux utilisateurs d'avoir une bonne idée de l'action s'y déroulant ou de l'ambiance dégagée. Toutefois le problème de la segmentation en scènes est délicat et leur utilisation pour l'indexation et la recherche d'information en est pour l'instant à ses débuts. La première étape incontournable est le découpage en plans, ensuite l'étude de l'organisation des plans permet de grouper les plans en scènes. Deux catégories se distinguent parmi les approches existantes. La première catégorie comprend les approches utilisant les algorithmes de regroupement. Les plans sont regroupés en fonction de leur similarité et éventuellement de contraintes temporelles [40]. Un graphe des transitions est ensuite construit et il permet de capturer la logique présente dans la séquence des plans. Dans la deuxième catégorie se trouvent les méthodes séquentielles qui regroupent au fur et à mesure les plans. Un ensemble de règle permet de définir si un plan appartient à la scène courante ou à une nouvelle scène [41] [42]. L'analyse des scènes offre l'opportunité d'avoir une indexation sémantique. Malheureusement le problème de la segmentation en scènes n'est actuellement pas correctement résolu dans le cas général et la plupart des méthodes d'indexation et de recherche de la vidéo reposent uniquement sur le découpage en plans. Par ailleurs le contenu d'une scène peut-être visuellement très varié et la notion de plan est toujours nécessaire pour identifier et caractériser les différents contenus. L'indexation des scènes repose donc sur les plans et plus particulièrement leurs images caractéristiques.

1.7.2.3 Segmentation en objet

La segmentation en objets n'est pas un problème facile, principalement parce que la définition des objets vidéo exige habituellement une interprétation sémantique de la scène. Il n'est généralement pas possible de définir de tels objets, sémantiquement significatifs, en termes de caractéristiques de bas niveau, tel que des paramètres de mouvement ou de couleur. Par conséquent, la segmentation et le suivi d'objets sémantiques dans une scène sans contrainte peuvent exiger l'intervention interactive de l'utilisateur. Cependant, dans quelques circonstances bien contraintes, des objets sémantiques peuvent être segmentés et suivis entièrement automatiquement. Par exemple, dans les systèmes de vidéo surveillance [38] [39] où la caméra est stationnaire, des objets dans la scène peuvent être extraits par des méthodes simples de soustraction et de détection de changement d'arrière plan.

1.7.2.4 Sélection d'images représentatives

Il est inutile d'entrer dans des calculs compliqués et longs pour toutes les images d'un plan afin d'extraire les caractéristiques visuelles. En effet, il serait par la suite impossible de conserver et d'utiliser cette information qui est par ailleurs redondante. Le processus de simplification de la vidéo continue donc par la sélection d'une ou plusieurs images représentatives des plans. Idéalement les images, appelées images clefs, doivent capturer le contenu sémantique du plan. Malheureusement les techniques de traitement de l'information ne sont pas assez avancées pour déterminer de telles images clefs. Les algorithmes utilisent donc les caractéristiques brutes obtenues sur les images (couleur, texture, mouvement). Lorsqu'un plan est statique, les images le composant sont souvent très similaires. Théoriquement il suffit alors de choisir l'image qui est la plus similaire aux autres. Malheureusement cette recherche exhaustive est difficilement réalisable en pratique. Les approches empiriques sélectionnent simplement la première, la dernière ou l'image médiane du plan. Yueting et al. [43] proposent une approche par regroupement des images similaires pour obtenir la ou les images représentatives du plan. Toutefois lorsqu'un plan est mouvementé, il est intéressant de sélectionner les images représentatives en fonction de l'intensité ou des variations du mouvement. Kobla et al. [44] présentent une approche dans le domaine compressé MPEG qui mesure le déplacement de la caméra dans le plan et découpe ce dernier en sous plans afin de limiter l'amplitude du mouvement. Wolf [45] utilise le flux optique pour détecter l'image avec l'activité la plus faible. Liu et al. [46] proposent une mesure de l'énergie du mouvement dans le domaine MPEG afin d'identifier les images clefs du plan. Une approche différente consiste à créer une mosaïque du plan (Irani and Anandan [47]). Cette approche est beaucoup plus rare car les mosaïques sont difficiles à construire dans le cas général.Une mosaïque est une unique image représentant l'ensemble de la scène. Elle est construite à partir des images du plan et fournit une représentation complète et compacte du fond. Les éléments mobiles sont décrits à part ainsi que leur trajectoire sur la mosaïque. Au final de nombreuses méthodes d'indexation et de recherche par le contenu visuel des vidéos sont très similaires aux méthodes employées sur les images fixes puisqu'elles se concentrent uniquement sur les images clefs. Toutefois la vidéo permet l'utilisation de caractéristiques propres comme le son, les sous-titres et les mouvements de la caméra et des objets. La difficulté sera alors de combiner cet ensemble de caractéristiques hétérogènes de manière efficace. Dans un premier temps nous concentrerons notre étude sur les caractéristiques calculées sur les images représentatives ou des séquences d'images.

1.8 Méthodes de description de vidéo

L'indexation des vidéos va s'appuyer sur l'extraction de caractéristiques qui peuvent être associées globalement aux différentes structures de la vidéo ou plus précisément à des images sélectionnées dans la vidéo.

1.8.1 Description du plan

De nombreuses méthodes ont été proposées dans la littérature pour capturer le contenu des plans et obtenir des signatures efficaces. La première partie traite de l'ensemble des méthodes d'indexation du contenu purement visuel. Ces méthodes sont généralement appliquées aux images représentatives des plans. Elles sont donc souvent similaires à celles que nous trouvons pour l'indexation d'images. La seconde partie traite de l'analyse du mouvement qui est

naturellement étudié sur l'ensemble du plan.

1.8.2 Description d'une image par des caractéristiques visuelles

Pour décrire le contenu visuel, de nombreuses méthodes sont proposées (Mandal et al. [48], Rui et al. [49]). Une première étape consiste à définir les caractéristiques requises en fonction du problème. Les différentes caractéristiques pouvant être extraites afin de décrire au mieux le contenu, peuvent être regroupées principalement dans trois catégories primaires : les caractéristiques de couleur, texture et structure, et une catégorie hybride. La description des couleurs est réalisée essentiellement par des histogrammes calculés dans différents espaces de couleur. La description de la texture est réalisée par une analyse fréquentielle ou l'étude d'occurrences. La description de la structure est réalisée par les coins, les angles, les contours et les formes. Les caractéristiques hybrides permettent de décrire conjointement les caractéristiques primaires. Elles se retrouvent principalement dans les domaines transformés, par exemple FFT, DCT et PCA. La deuxième étape consiste à définir les régions de l'image dans lesquelles les données vont être extraites. Selon l'application, il sera suffisant de décrire l'image comme une unique entité. D'autres applications plus complexes peuvent nécessiter une description locale du contenu faisant appel à une détection des régions ou des objets. Des compromis doivent être établis entre l'objectif de l'application, la complexité de la représentation et les temps de calcul requis pour le traitement des données.

1.8.2.1 A partir de la texture

La texture est importante pour caractériser les motifs présents dans l'image, cependant elle n'a aucune définition précise. Quatre types d'approches se distinguent [50] : les approches statistiques, géométriques, spectrales et par modélisation. Dans la première catégorie nous trouvons les matrices de co-occurrence [51]. Dans la seconde nous trouvons les descripteurs de Tamura [52] qui caractérisent la granularité, la direction et le contraste. Dans la troisième nous trouvons les ondelettes avec les filtres de Gabor [53] [54] qui permettent de capturer les fréquences et les directions principales. Finalement dans la dernière catégorie nous trouvons la décomposition de Wold [55] qui caractérise la périodicité, la direction et le désordre ; ainsi que les modèles autorégressifs simultanés et multi résolution [56] (ou « multiresolution simultaneous autoregressive models : MSAR ») qui modélisent la texture à différents niveaux de granularité en fonction du voisinage des pixels. Malgré les contraintes spatiales qui peuvent être imposées par certaines méthodes, la notion de région et idéalement d'objet fait toujours défaut.

1.8.2.2 A partir des régions

Les systèmes basés sur les régions composant une image tentent de capturer le contenu d'une manière qui reflète le comportement humain. Pour cela, l'image est segmentée en régions homogènes puis les algorithmes travaillent au niveau de la granularité des régions. Ainsi les propriétés locales de l'image sont analysées. Les systèmes qui proposent d'indexer les régions des images sont rares malgré leurs attraits. En effet deux barrières s'opposent à leur développement. La première et principale est la segmentation en objets. Ce délicat problème est actuellement loin d'être résolu dans le cas général. Les algorithmes de segmentation en régions sont nombreux mais ne parviennent pas toujours à discerner les objets. Par ailleurs ils sont souvent très sensibles et il est difficile d'obtenir une segmentation homogène entre

plusieurs images. La seconde barrière est le coût de l'analyse, du stockage et des calculs de similarité. Les opérations d'indexation et de recherche se font sur les régions qui sont dissociées de l'image à laquelle elles appartiennent [57] [58]. Les méthodes de la seconde catégorie utilisent l'ensemble des régions composant une image pour effectuer l'indexation et la recherche. Deux mesures de similarité sont communément employées. La méthode IRM (Integrated Region Matching) [59] permet de mettre en correspondance toutes les régions de deux images. La mesure EMD (Rubner et al. [60]) Earth Mover's Distance permet de mesurer le coût nécessaire pour transformer un ensemble de régions en un deuxième ensemble de régions. Trois autres représentations des images segmentées ont également été proposées. La représentation de l'image et de ses régions peut être faite par des graphes [61] qui permettent une comparaison simultanée des propriétés et de l'agencement des régions. Elle peut également être faite par des chaînes qui décrivent les relations entre les régions [62]. Finalement elle peut être réalisée par un vecteur de dénombrement des régions d'un dictionnaire [63]. Le découpage des images en régions permet aussi de capturer de nouvelles caractéristiques concernant les structures des images comme nous allons le voir.

1.8.3 Utilisation des informations du flux de compression

L'extraction des caractéristiques des plans à partir du flux de compression a montré un grand intérêt de la part de nombreux chercheurs. Le but principal est de limiter les temps de calcul en utilisant l'information existante en l'état. Le flux vidéo permet l'accès direct aux coefficients DCT des macros blocs qui fournissent à la fois une information locale de couleur et de texture. Il permet également l'accès à une information de mouvement qui peut s'avérer particulièrement riche [45] [64]. Il est alors possible de traiter toutes les images de la vidéo en se contentant de ces coefficients. Le mouvement est une information riche qui renseigne sur l'activité d'un plan et celle de ses objets. A partir de la séquence des images formant un plan, le mouvement de la caméra peut être estimé ; ensuite le mouvement des objets peut être déterminé. Contrairement aux problèmes de vision comme la localisation ou la reconstruction, l'indexation ne requiert pas nécessairement une grande précision dans l'estimation du mouvement. Le choix de la méthode de calcul ne sera donc pas abordé puisque de nombreuses méthodes conviennent. Les deux méthodes les plus répandues sont (Horn and Schunck [65], Lucas and Kanade [66]) ; cependant la plupart des méthodes utilisent directement les vecteurs mouvements fournis par les normes de compression (MPEG). L'objectif est alors d'éviter le lourd calcul du flux optique pour chaque image en utilisant directement le mouvement déjà calculé par les algorithmes de compresssion et accessible dans les données compressées. . Une première approche de l'indexation par le mouvement utilisant directement l'information du flux compressé est présentée par Kobla et al. [45], un vecteur caractéristique qui représente la direction quantifiée du mouvement pour chaque macro bloc du flux MPEG. Le mouvement peut également être indexé par un histogramme de l'ensemble des vecteurs disponibles [67] ou des attributs caractéristiques [68]. Les paramètres obtenus peuvent alors servir à l'indexation ou être classés dans les catégories zoom, rotation, translation verticale et horizontale pour une indexation sémantique. Dans [69], un système d'indexation basé sur la trajectoire des objets est proposé. Dagtas et al. [70] présentent une approche d'indexation des trajectoires accompagnée d'une méthode de recherche des trajectoires qui a l'avantage d'être invariante dans l'espace et robuste aux changements d'échelle. DeMenthon et Doermann [71] proposent une description spatiale et temporelle. Une segmentation spatio-temporelle est d'abord réalisée. Les régions obtenues dans chaque image sont ensuite indexées avec l'objet auquel elles appartiennent. La recherche permet alors de retrouver les objets aux mêmes propriétés visuelles et de mouvement. Syeda-Mahmood [72] propose une modélisation en 3D de l'objet et de son

évolution dans le temps. Cette modélisation permet ensuite de retrouver les objets identiques et au mouvement similaire sous différents points du vue.

1.9 Indexation et recherche de vidéos par le contenu dans le domaine médical

1.9.1 Les méthodes

Dans le domaine médical, les méthodes utilisées peuvent être regroupées dans 2 catégories :

- Les méthodes basées sur la segmentation de formes d'intérêt (JSEG, chaînes de markoff, contours actifs, etc...) telles que présentées dans [25] [28], associées à l'extraction de descripteurs bas niveau connus pour bien caractériser les zones d'intérêt (histogramme de couleur, texture, forme, etc) qui sert alors d'index aux vidéos.
- Les méthodes basées sur l'extraction de mouvement entre les images (flot optique, domaine de compression, etc) [34,35], puis la caractérisation de cette information (quanification) afin de construire des signatures de vidéos

Il faut noter que les méthodes de détection de plan sont rarement utilisées dans les applications médicales. Les vidéos auxquelles nous nous intéressons dans cette étude sont acquises à l'aide d'une seule caméra, contrairement aux séquences vidéos obtenues après montage de plusieurs plans fournis par des caméras en mouvement ou la caractérisation consiste dans un premier temps à détecter les différents plans puis extraire les éléments pertinents au sein des plans considérés. Nous rappelons qu'un plan est une succession d'images obtenues par une acquisition unique et continue à partir d'une caméra. Les éléments qui peuvent nous intéresser sont liés à l'environnement, les changements peuvent intervenir au sein de la séquence et les indices temporels qui y sont associés, aux régions (zones d'intérêt) se déplacent au sein de leur environnement et interagissent soit entre eux, soit avec l'environnement. A partir de ce constat, nous avons décidé de nous intéresser aux indices relatifs aux régions. En considérant successivement ceux liés à leurs mouvements (ou déplacements), ces régions peuvent être assimilées à une information aussi bien de bas niveau (niveau de gris, histogramme, couleur, ...etc.) ou plutôt une information de haut niveau (déplacement, texture, forme, ... etc).

Le moyen le plus courant d'analyser les régions présentes dans une séquence vidéo (et donc d'en extraire des indices) consiste à étudier leur position au cours du temps, donc à caractériser leur mouvement. Ce problème, qui est un problème de suivi d'objet, a donné lieu à de nombreux travaux (voir section précédente). Le suivi d'objet ou de région est formulé de la façon suivante : connaissant la position d'une région R dans les images précédentes, on souhaite déterminer sa position dans l'image courante. Pour rechercher la région dans l'image courante, l'approche la plus utilisée requiert l'identification des caractéristiques de l'objet (mouvement, forme, couleur, texture, etc.) puis la localisation de ces caractéristiques dans l'image à analyser. Ces caractéristiques peuvent être obtenues en recherchant dans l'image courante les caractéristiques les plus similaires à celles trouvées dans l'image précédente.

1.9.2 Notre choix

Il est difficile de trouver des méthodes robustes à toutes les situations et suffisamment rapides pour segmenter les formes d'intérêt, les objets caractéristiques, dans des vidéos chirurgicales en temps réel. Nous n'avons non plus qu'un seul plan dans nos vidéos. Nous avons donc choisi de nous intéresser principalement aux mouvements des régions dans les séquences vidéos

pour les caractériser. Nous avons déjà vu que de nombreuses méthodes existent pour extraire cette information (voir section précédente). Nous avons utilisé la dernière norme de compression (MEPG4/H.264) pour extraire cette information et d'autres indices pour caractériser les vidéos. L'intérèt d'utiliser directement la norme est qu'il existe des cartes éléctroniques de compression qui génèrent le flux vidéo compressé en temps réel. Nous n'avons alors pas à travailler directement sur la séquence d'images et profitons des données et paramètres calculés dans la norme de compression, données qui contiennent bien toute l'information utile et concise pour caractériser les vidéos.

1.10 Conclusion

Pour les raisons évoquées précédemment, nous allons utiliser les informations du flux de données compressées, essentiellement les informations liées au codage des déplacements entre blocs de codage, pour caractériser et indexer nos séquences vidéos.

Notre travail portera principalement sur les points suivante :

- développement de nouveaux paramètres ou signatures, pertinents et rapides à calculer, qui caractérisent bien les vidéos, à partir de la norme de compression H264/AVC.
- étude de nouvelles métriques de similitude adaptées à ces signatures, prenant en compte les contraintes de temps réel, pour une recherche rapide et efficace.

Dans le chapitre 2, nous présentons les principes de compression des séquences vidéos, et plus particulièrement la norme H264/AVC. Cela nous permettra de justifier l'utilisation de cette norme et mieux comprendre les éléments que nous utiliserons pour construire nos signatures.

Bibliographie

[1] C. LeBozec, 'Unified modeling language and design of a case-based retrieval system in medical imaging', presented at the Proceedings of the Annual Symposium of the American Society for Medical Informatics (AMIA), Nashville, TN, USA, 1998.

[2] Leake D. B., editor., Case-Based Reasoning : Experiences, Lessons, and Future Directions, AAAI Press/MIT Press, Menlo Park, CA, 1996.

[3] R.C. Schank and R.P. Abelson. Scripts, Plans, Goals, and Understanding. Hills-dale, N.J. : Erlbaum, 1977.

[4] I. Bichindaritz and C. Marling. Case-based reasoning in the health sciences : What's next Artificial Intelligence in Medicine, 36(2) :127 135, January 2006.

[5] Aditi Roy, Shamik Sural, Jayanta Mukherjee, Arun K. Majumdar, State-Based Modeling and Object Extraction From Echocardiogram Video, IEEE Transactions on Information Technology in Biomedicine, Volume 12 Issue 3, May 2008 Page 366-376

[6] I. Bichindaritz, Case-based reasoning in the health sciences : What is next , Artificial Intelligence in Medicine, vol. 36, pp. 127 135, january 2006.

[7] J. L. a. J. Z. Wang., Automatic linguistic indexing of pictures by a statistical modeling approach, IEEE Transactions on Pattern Analysis and Machine Intelli- gence,, vol. 25, pp. 1075 1088, September 2003.

[8] T. Kato., Database architecture for content-based image retrieval, presented at the Proc. SPIE, San Jose, CA, USA 1992.

[9] R. P. A. Pentland, and S. Sclaroff, "Photobook : Content-based manipulation of image databases" International Journal of Computer Vision, vol. 18, pp. 233-254, June 1996.

[10] M. T. C. Carson, S. Belongie, J. M. Hellerstein, and J. Malik, Blobworld : A sys- tem for region-based image indexing and retrieval, presented at the International Conference On Visual Information Systems (VISUAL 99), 1999.

[11] L. L. a. S. K. Mitra, Unsupervised segmentation of color images based on kmeans clustering in the chromaticity plane, presented at the IEEE Workshop on Content- based Access of Image and Video Libraries (CBAIVL 99), 1999.

[12] C. E. B. C.R. Shyu, A.C. Kak, A. Kosaka, A. Aisen, and L. Broderick. , Local versus global features for content-based image retrieval, presented at the In IEEE Workshop on Content-Based Access of Image and Video Libraries, 1998.

[13] J. Y. L. M.K. Markey, G.D. Tourassi, and C.E. Floyd Jr, Self-organizing map for cluster analysis of a breast cancer database. Artificial Intelligence in Medicine, vol. 27 pp. 113-127, 2003.

[14] E. P. a. C. Faloutsos, Similarity searching in medical image databases, IEEE Trans. Knowledge and Data Eng, vol. 9, pp. 435 447, June 1997.

[15] F. M. V. H.D. Tagare, C.C. Jaffe, and J.S. Duncan, Arrangement - a spatial relation between parts for evaluating similarity of tomographic section, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 17, pp. 880 983, september 1995

[16] M. K. W. Qian, L.P. Clarke, H.D. Li, D. Venugopal, D.S. Song, and L.P. Clark, Treestructured wavelet transform segmentation of microcalcifications in digital mammography, Medical Physiology, vol. 22, pp. 1247 1254, 1995.

[17] Y. L. a. F. Dellaert, Classification driven medical image retrieval, presented at the the Image Understanding Workshop, 1998.

[18] W. Hu, D. Xie, Z. Fu, W. Zeng, and S. Maybank, Semantic-based surveillance video retrieval, IEEE Trans Image Process, vol. 16, no. 4,pp. 11681181, 2007.

[19] R. T. Collins, A. J. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, and O. Hasegawa. A system for video surveillance and mon- itoring. Tech. report CMU-RI-TR-00-12, Robotics Institute, Carnegie Mellon uni- versity, May 2000.

[20] C. Cedras and M. Shah. Motion-based recognition : a survey. Image Vision Com- puting, 13(2) :129 155, March 1995.

[21] D. M. Gavrila. The visual analysis of human movement : a survey. Computer Vision and Image Understanding, 73(1) :82 98, January 1999

[22] Peijiang Yuan, Bo Zhang, Jianmin Li : Multi-modal Information Retrieval for Content-based Medical Image and Video Data Min-ing. IMAGAPP 2009 : 83-86

[23] Z. Droueche, M. Lamard, G. Cazuguel, G. Quellec, C. Roux and B. Cochener, Content-Based Medical Video Retrieval Based on Region Motion Trajectories IFMBE Proceedings, 2012, Volume 37, Part 1, Part 6, 622-625, DOI : 10.1007/978- 3-642-23508-5-161

[24] Barbara Andre, Tom Vercauteren, Anna M. Buchner, Michael B. Wallace, and Nicholas Ayache. A Smart Atlas for Endomicroscopy using Automated Video Re- trieval. Medical Image Analysis, 15(4) :460-476, August 2011

[25] L. Zelnik-Manor and M. Irani. Event-based analysis of video. In CVPR, pages II :123130, 2001.

[26] S. Giannarou and G.Z. Yang, Content-based surgical workflow representation using probabilistic motion modeling, in LNCS Medical Imaging and Augmented Reality, vol. 6326, 2010, pp. 314323

[27] H. Liao, N. Hata, S. Nakajima, M. Iwahara, I. Sakuma, T. Dohi, Surgical Navi- gation by Autostereoscopic Image Overlay of Integral Videography, IEEE Trans. Inform. Technol. Biomed., Vol.8 No.2, pp.114-121, June 2004

[28] Y. Cao, D. Liu, W. Tavanapong, J. Wong, J. Oh, and P. de Groen, Computer- aided detection of diagnostic and therapeutic operations in colonoscopy videos, IEEE Trans Biomed Eng, vol. 54, no. 7, pp. 12681279, 2007

[29] T. Blum, et al., Modeling and Online Recognition of Surgical Phases Using Hidden Markov Models, MICCAI 2008, Part II LNCS, Vol. 5242, pp. 627-635.

[30] Florent Lalys, Laurent Riffaud, David Bouget, Pierre Jannin : A Framework for the Recognition of High-Level Surgical Tasks From Video Images for Cataract Surgeries. IEEE Trans. Biomed. Engineering 59(4) : 966-976 (2012)

[31] A. M. Cano, F. Gaya, P. Lamata, P. Sanchez-Gonzalez, and E. J. Gomez, Laparo- scopic tool tracking method for augmented reality surgical applications, in LNCS, vol. 5104, 2008, pp. 191196

[32] S. Seshamani, W. Lau, and G. Hager, Real-time endoscopic mosaicking,in MIC- CAI, no. 9, 2006, pp. 355363

[33] U. Gargi, R. Kasturi, and S. H. Strayer, Performance characterization of video- shot change detection methods, IEEE Trans. Circ. Syst. for Video Tech., vol. 10, pp. 1-13, Feb. 2000.

[34] R. Lienhart, Reliable transition detection in videos : A survey and practitioner's guide, Int. J. Image Graph., vol. 1, pp. 469-486, Aug. 2001.

[35] A. Hanjalic, Shot-boundary detection : Unraveled and resolved IEEE Trans. Circ. Syst. for Video Tech., vol. 12, pp. 90-105, Feb. 2002.

[36] Philippe Joly, Hae-Kwang Kim. Efficient Automatic Analysis of Camera Work and Microsegmentation of Video Using Spatio-Temporal Images. Signal Processing : Image Communication, Elsevier, Eurasip, Amsterdam. 1996.

[37] Aigrain Ph, Joly Ph, Longueville V. Medium Knowledge-Based Macro- Segmenta- tion of Video into Sequences. In M. Maybury (Ed.) (pp. 5-16), IJCAl 95 - Workshop on Intelligent Multimedia Information Retrieval. Montreal, August 19, 1995.

[38] J. D. Courtney, Automatic video indexing via object motion analysis, Pattern Recognition, vol. 30, no. 4, pp. 607-626, April 1997.

[39] G. L. Foresti, L. Marcenaro, and C. S. Regazzoni, Automatic detection and index- ing of video-event shots for surveillance applications, IEEE Trans. Multimedia, vol. 4, no. 4, pp. 59-471, Dec. 2002.

[40] Minerva M. Yeung and Boon-Lock Yeo. Time-constrained clustering for segmenta- tion of video into story unites. In Proceedings of the IEEE International Conference on Pattern Recognition, volume 3, pages 375 380, 1996.

[41] Alan Hanjalic, Reginald L. Lagendijk, and Jan Biemond. Automated high-level movie segmentation for advanced video-retrieval systems. IEEE Transactions on Circuits and Systems for Video Technology, 9(4) :580 588, 1999.

[42] Wallapak Tavanapong and Junyu Zhou. Shot clustering techniques for story brows- ing. IEEE Transactions on Multimedia, 6(4) :517 527, august 2004.

[43] Zhuang Yueting, Rui Yong, T.S. Huang, and S. Mehrotra. Adaptive key frame extraction using unsupervised clustering. In Proceedings of the IEEE International Conference on Image Processing, volume 1, pages 866870, 1998.

[44] Vikrant Kobla, David S. Doermann, King-Ip Lin, and Chritos Faloutsos. Com- pressed domain video indexing techniques using DCT and motion vector informa- tion in MPEG video. In Proceedings of SPIE conference on Storage and Retrieval for Image and Video Databases, volume 3022, pages 200 211, february 1997.

[45] W. Wolf. Key frame selection by motion analysis. In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, volume 2, pages 1228 1231, 1996.

[46] Tianming Liu, Hong-Jiang Zhang, and Feihu Qi. A novel video key-frame extrac- tion algorithm based on perceived motion energy model. IEEE Transactions on Circuits and Systems for Video Technology, 13(10) :1006 1013, october 2003.

[47] M. Irani and P. Anandan. Video indexing based on mosaic representation. IEEE Transaction on PAMI, 86(5) :905 921, 1998.

[48] M.K. Mandal, F. Idris, and S. Panchanathan. A critical evaluation of image and video indexing techniques in the compressed domain. Image and Vision Computing Journal, 17(7) :513 529, may 1999

[49] Y. Rui, T. Huang, and S. Chang. Image retrieval : current techniques, promising directions and open issues. Journal of Visual Communication and Image Repre- sentation, 10(4) :39 62, April 1999.

[50] Mihran Tuceryan and Anil K. Jain. The Handbook of Pattern Recognition and Computer Vision (2nd Edition), chapter Texture Analysis, pages 207 248. World Scientific Publishing Co., 1998.

[51] Robert M. Haralick. Statistical and structural approaches to texture. Proceedings of the I.E.E.E., 67(5) :786 804, May 1979.

[52] T. Yamawaki H. Tamura, S. Mori. Textural features corresponding to visual per- ception. IEEE Transaction on Systems, Man and Cybernetics, 8 :460 482, 1978.

[53] M. R. Turner. Texture discrimination by gabor functions. Biol. Cybern., 55(2- 3) :71 82, 1986.

[54] B. S. Manjunath and W. Y. Ma. Texture features for browsing and retrieval of image data. IEEE Trans. Pattern Anal. Mach. Intell., 18(8) :837 842, 1996.

[55] F. Liu and W. Picard. Periodicity, directionality, and randomness : Wold features for image modeling and retrieval. IEEE Transaction Pattern Analysis and Machine Intelligence, 18 : 722 733, 1996

[56] Jianchang Mao and Anil K. Jain. Texture classification and segmentation us- ing multiresolution simultaneous autoregressive models. Pattern Recognition, 25(2) :173 188, 1992.

[57] J. Fauqueur and N. Boujemaa. Region-based image retrieval : Fast coarse segmentation and fine color description. Journal of Visual Languages and Computing, 15(1) :6995, february 2004. BIBLIOGRAPHIE 19

[58] H. Foroosh. Pixelwise-adaptive blind optical flow assuming nonstationary statistics. IEEE

[59] Transactions on Image Processing, 14(2) :222230, february 2005. James Z.Wang, Jia Li, and GioWiederhold. SIMPLIcity : Semantics-sensitive integrated matching for picture libraries. IEEE Transactions on Pattern Analysis and Machine Intelligence, 23(9) :947963, 2001

[60] Y. Rubner, C. Tomasi, and L. J. Guibas. A metric for distributions with applications to image databases. In Proceedings of the IEEE International Conference on Computer Vision, pages 5966, january 1998.

[61] J. Matas, R. Marik, and J. Kittler. On representation and matching of multi-coloured objects. In Proceedings of the IEEE International Conference on Computer Vision, pages 726732, 1995.

[62] Suh-Yin Lee and Fang-Jung Hsu. Spatial reasoning and similarity retrieval of images using 2d c-string knowledge representation. Pattern Recogn., 25(3) :305318, 1992.

[63] Joo-Hwee Lim. Learning visual keywords for content-based retrieval. In IEEE International Conference on Multimedia Computing and Systems, volume 2, pages 169173, 1999.

[64] Cuneyt Taskiran, Jau-Yuen Chen, Alberto Albiol, Luis Torres, Charles A. Bouman, and Edward J. Delp. Vibe : a compressed video database structured for active browsing and earch. IEEE Transactions on Multimedia, 6(1) :103 118, february 2004.

[65] B. Horn and B. Schunck. Determining optical flow. Artificial Intelligence, 17 :185 203, 1981.

[66] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In International Joint Conference on Artificial Intelligence, pages 674 679, 1981

[67] Deng Yining and B.S. Manjunath. Content-based search of video using color, tex- ture, and motion. In Proceedings of the IEEE International Conference on Image Processing, volume 2, pages 534 537, 1997.

[68] E. Ardizzone and M. La Cascia. Video indexing using optical flow field. In Pro- ceedings of the IEEE International Conference on Image Processing, volume 3, pages 831 834, 1996.

[69] Shih-Fu Chang,W. Chen, H.J. Meng, H. Sundaram, and Di Zhong. A fully au- tomated contentbased video search engine supporting spatiotemporal queries. In IEEE Transactions on Circuits and Systems for Video Technology, volume 8, pages 602 615, 1998.

[70] S. Dagtas, W. Al-Khatib, A. Ghafoor, and R.L. Kashyap. Models for motion- based video indexing and retrieval. IEEE Transactions on Image Processing, 9(1) :88 101, January 2000.

[71] Daniel DeMenthon and David Doermann. Video retrieval using spatio-temporal descriptors. In Proceedings of the ACM International Conference on Multimedia, pages 508 517, 2003.

[72] Tanveer Syeda-Mahmood. Retrieving actions embedded in video. In Proceedings of the ACM International Conference on Multimedia, pages 513 522, 2002.

CHAPITRE 2

LE CODAGE VIDEO ET LA NORME H.264/AVC

Comme nous l'avons déjà mentionné dans l'introduction, l'objectif de cette thèse est de proposer un système d'aide au geste chirurgical en peropératoire utilisant la CBVR.

La CBVR doit traiter des quantités d'informations importantes dans des temps admissibles compatibles avec le travail préopératoire. Pour cela, nous nous somme intéressés aux informations contenues dans le domaine compressé des vidéos : les données compressées contiennent toute l'information nécessaire lorsque les taux de compression ne sont pas trop élevés, ce qui est le cas pour des vidéos médicales, et l'extraction des données est rapide, ne requérant pas une décompression totale de la vidéo.

La compression de vidéo consiste à réduire la quantité de données nécessaires pour la stocker ou la transmettre, tout en cherchant à minimiser les phénomènes de perte de qualité. Elle s'appuie en particulier sur le fait que les variations à l'intérieur d'une séquence, causées par exemple par des mouvements d'objets/régions par rapport à la caméra ou des changements d'illumination, sont continues dans la majorité des cas. Les données contenues sont par conséquent imprégnées de fortes redondances, qu'il est possible de retirer pour représenter la scène en utilisant une quantité réduite d'informations. De plus, l'oeil humain ne perçoit qu'une partie des informations présentes dans les images, et il est ainsi possible de retirer les zones les moins pertinentes en limitant l'impact sur la qualité ressentie. L'utilisation conjointe de la suppression des redondances et des informations les moins visibles permet alors de condenser de manière significative les données à représenter.

Dans ce chapitre, nous allons présenter le principe de fonctionnement des codeurs vidéo, et plus particulièrement le codeur H.264/AVC sur lequel nous nous appuierons pour créer des signatures de vidéo pour la CBVR. Nous expliquons chacune des étapes de codage à l'aide d'un schéma basique de codage de cette norme. L'objectif principal est d'exposer brièvement le cadre global de la compression vidéo et de faire ressortir les éléments qui permettront de développer des signatures associées aux principales méthodes de compression et/ou dans les différentes étapes de ces méthodes. Les concepts présentés permettront aussi d'expliquer certains résultats de performance de retrouvaille associées aux signatures, que nous proposerons dans la suite.

Le chapitre commence par une brève discussion sur les différentes codeurs vidéo existants, suivi d'une description succincte des principales étapes du codage H.264/AVC notamment la définition des différentes mesures d'erreur et les approches d'évaluation de la qualité de la compression qui ont amené à privilégier cette norme. Nous terminons le chapitre par un bilan

de l'état de l'art et quelques comparaisons, qui ne concernent pas directement nos objectifs mais donnent des résultats pertinents.

2.1 Historique

Une présentation détaillée de l'évolution historique des codeurs vidéo est donnée dans [1]. La manière la plus simple de réaliser un codage vidéo consiste à coder chaque image du flux par le biais d'un codeur d'images fixes, par exemple JPEG [2]. Dans ce type de schéma, seules les redondances spatiales des images sont utilisées pour comprimer la vidéo. Cependant, l'idée d'exploiter la corrélation temporelle entre les images successives d'une séquence d'images apparaît dès 1929 [3]. On se préoccupait déjà de réduire la quantité d'information à transmettre. Dans cette approche, seules les différences entre les images successives doivent être transmises au récepteur (prédiction temporelle entre images). La toute première norme proposée par l'ITU-T (International Telecommunications Union-Telecommuniction), H.120 [4], met en oeuvre cette idée. Dans cette norme, les images sont découpées en blocs. Les blocs identiques à leurs correspondants dans l'image précédente ne sont pas codés et les autres sont traités par prédiction spatiale (dans l'image elle même).

Les codeurs de type H.120 sont cependant inefficaces en cas de mouvement d'ensemble de la caméra vis-à-vis de la scène filmée. Pour résoudre ce problème, la solution a été d'exploiter une partie des informations contenues dans l'image précédente pour prédire les blocs de l'image courante et de transmettre des données permettant de corriger cette prédiction [5]. Ce principe a donné naissance aux codeurs vidéo hybrides, dont le nom provient du fait qu'ils utilisent deux techniques de réduction de redondances : d'une part, une prédiction temporelle, d'autre part, une transformation des résidus de prédiction (transformation de la différence entre l'image courante et l'image prédite). Cette structure de base a été formalisée par l'ITU-T dans la norme H.261 [6]. Les normes ultérieures, MPEG-1 [7], MPEG-2 [8], H.263 [9], MPEG-4 Part 2 Visual [10] et H.264/AVC [11] ont essentiellement repris et amélioré cette structure de base de codage hybride.

2.2 Les outils élémentaires pour la compression

Quel que soit le type de codeur (voix, musique, image fixe, vidéo), un certain nombre d'outils communs permettent de réaliser la compression du signal. Ces outils, brièvement décrits ci-dessous et illustrés sur la figure 2.1, sont largement détaillés dans [12].

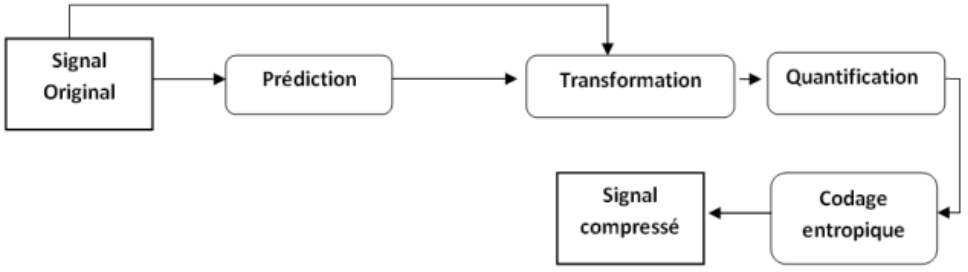


Figure 2.1 — Schéma global d'un codeur pour la compression de signaux

2.2.1 La prédiction

Ce mécanisme a pour fonction de prédire les valeurs du signal d'entrée pour les instants à venir. Généralement, cette estimation est basée sur les informations des échantillons déjà

encodés du signal. La différence entre les échantillons du signal original et leur prédiction permet d'obtenir les résidus de prédiction qui demandent moins de bits de codage, et rendent plus efficace la compression.

2.2.2 La transformation

Ce procédé consiste à projeter les résidus de prédiction dans une base permettant de réduire la corrélation statistique entre les échantillons des résidus de prédiction. Cette opération permet de rassembler l'énergie du signal sur un faible nombre d'échantillons. Une décorrélation optimale des données est obtenue à partir de la transformée de Karhunen-Loève (Karhunen-Loève Transform ou KLT) qui projette le signal sur les vecteurs propres de sa matrice de covariance. Cependant, cette transformation est complexe et les codeurs traditionnels intègrent plutôt une décomposition en sous-bandes fréquentielles. Dans cette catégorie, la transformée en cosinus discrète (Discrete Cosine Transform ou DCT) est de loin la plus utilisée à la fois pour sa simplicité algorithmique et ses performances [13].

2.2.3 La quantification

Ce procédé permet de faire une approximation des échantillons du signal transformé afin de réduire la quantité d'information à transmettre. Souvent, la précision de cette approximation est contrôlée à l'aide d'un pas représentant la plus petite valeur absolue représentable, mais également l'incrément entre deux valeurs successives à la sortie du quantificateur. Parmi tous les outils de compression, la quantification est la seule opération à être irréversible, induisant des pertes d'information. L'objectif de tout codeur est de minimiser la perte d'information résultante sous une contrainte de débit en sortie du codeur.

2.2.4 Le codage entropique

Ce mécanisme permet de représenter efficacement les symboles issus du quantificateur en prenant en compte leur fréquence d'apparition. Ainsi, un mot de code de longueur variable (Variable Length Coding ou VLC) est associé à chaque symbole en fonction de la statistique de la source. Les mots de code les plus courts sont attribués aux symboles fréquents, et inversement, des mots de code plus longs sont réservés aux symboles les moins probables. Les codeurs actuels utilisent ce procédé pour encoder les paramètres de compression, le plus connu étant le codage de Huffman.

2.3 La norme H.264/AVC

2.3.1 Description du schéma global de codage de la norme H.264/AVC

La norme H.264/AVC [14] est un codeur vidéo en boucle fermée, tout comme l'ensemble de ses prédécesseurs. En effet, les informations déjà codées sont utilisées pour le codage de l'image courante. Le schéma global de codage de cette norme est représenté dans la figure 2.2. La séquence en entrée (en haut à gauche dans le schéma) est une succession d'images (matrices de pixels). Chaque image est découpée en "tranches". Une tranche (slice en anglais) est une partie de l'image ou l'image entière en fonction des paramètres d'entrée du codeur. Ces tranches sont découpées en macroblocs (blocs de taille 16x16). Le macrobloc est l'unité

de codage dans la norme. Chaque macrobloc est codé soit en prédiction spatiale (boite 1), soit en prédiction temporelle (Compensation de mouvement, boite 12), chacun de ces codages engendre plusieurs résidus (cf 1.3.8), qui sont comparés dans le module de décision (boite 2). Le système de codage (inter ou intra) qui donne la meilleure possibilité de codage, par rapport à un critère débit-distorsion est alors sélectionné. Ce résidu est dé-corrélé avec la transformée en cosinus discret (DCT) (boite 3). Il est ensuite quantifié (boite 4) et les coefficients sont ensuite envoyés vers le codeur entropique sans perte (boite 10) qui produit le flux binaire.

Les résidus transformés et quantifiés sont ensuite dé-quantifiés (boite 5) et on leur applique la transformée inverse (boite 6) à l'intérieur du codeur. Aux blocs ainsi générés, on applique la prédiction inverse (7), en utilisant le prédicteur sélectionné dans le module de décision (boite 2) (le meilleur prédicteur Intra ou Inter). Ensuite, un filtre anti-blocs est appliqué sur l'image reconstruite, permettant d'éliminer certaines dégradations produites par le module de quantification (boite 4). Ce filtre lisse les images de référence en bordure des blocs. Enfin, les macroblocs et slices décodés sont stockés en mémoire (boite 9). Les blocs décodés de l'image courante, stockés dans ce module, sont utilisés pour le calcul des prédicteurs Intra. De même, les images précédemment décodées, où l'effet de bloc a été lissé, sont utilisées pour le codage Inter. En effet, le mouvement entre le bloc courant et ces images est estimé (boite 11). Le prédicteur obtenu par cette estimation de mouvement servira pour la prédiction Inter que l'on appelle généralement compensation de mouvement (boite 12). Afin que le décodeur soit capable de retrouver ce prédicteur Inter, un vecteur correspondant au mouvement entre le bloc courant et le prédicteur est transmis au décodeur. Les vecteurs issus du codage Inter sont prédits (boite 13) puis envoyés dans le codeur entropique. Ces vecteurs prédits sont insérés dans le flux binaire.

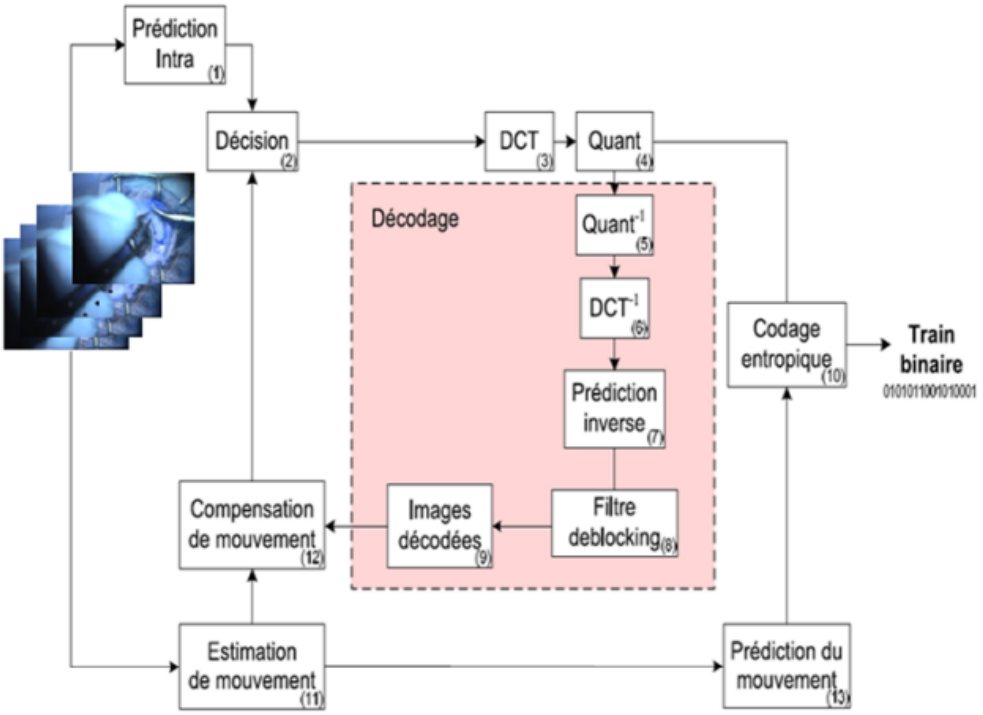


Figure 2.2 — Schéma global d'un codeur H.264/AVC

Dans la norme, on distingue trois types d'image : I, P, B. Dans une image I, tous les macroblocs sont prédits en utilisant le mode Intra. De manière idéale, une image I devrait intervenir lors d'un changement de scène, c'est-à-dire lorsque les redondances temporelles entre les images sont faibles. Dans la pratique, les programmes de détection de changement de scène sont délicats à implémenter et les codeurs insèrent généralement une image I à intervalles réguliers. Dans une image P, chaque macrobloc est prédit en utilisant soit le mode Intra, soit le mode Inter. Lorsque le mode Inter est actif chaque macrobloc est associé à une seule image de référence. Contrairement aux images P, les macroblocs d'une image B construits avec le mode Inter peuvent s'appuyer sur deux différents types d'image. Les 3 types d'images sont regroupés pour former des séquences (Group Of Pictures ou GOP). Un GOP débute par une image I et contient ensuite une succession d'images P et B. La structure classique d'un GOP est illustrée sur la figure 2.3.

Les GOPs sont indépendants entre eux. Cette technique permet au décodeur de se resynchroniser sur le flux dans le cas d'une transmission avec pertes.

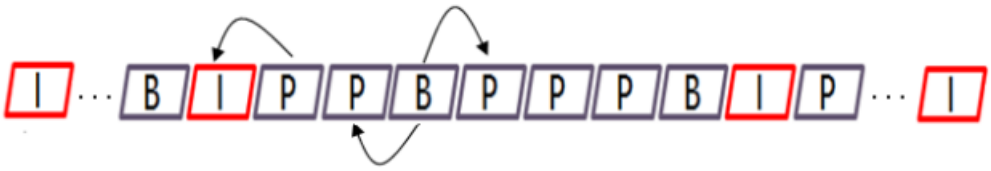


Figure 2.3 — Structure d'un GOP et dépendance entre images

Dans la suite, nous présentons en détail le principe de fonctionnement des différents outils intervenant dans le codeur

2.3.2 Prédiction Inter

Dans le cas de la prédiction Inter [15], les macroblocs d'une image (image déjà compressée) sont prédits à partir des échantillons d'une image de référence précédemment encodée. Afin de réaliser précisément cette opération, chaque macrobloc peut être divisé en partitions de taille variable 16x16, 16x8, 8x16, 8x8, et chaque bloc 8x8 peut avoir un partitionnement 8x8, 8x4, 4x8 ou 4x4. Les partitions d'un macrobloc et d'un sous-macrobloc sont illustrées sur la figure 2.4. Cette décomposition pyramidale permet d'isoler les différents objets composant une image et de s'adapter à leurs caractéristiques (sens de déplacement, vitesse). Un vecteur de mouvement est associé à chaque partition d'un macrobloc. Ce vecteur de déplacement spécifie l'écart spatial entre la partition courante de l'image actuelle et sa meilleure représentation dans l'image de référence basée sur le critère de fidélité. Pour le bloc courant, ce processus a pour but de rechercher dans les images précédemment encodées, un bloc de même taille représentant le même objet ou une même partie de l'objet. Ce processus de recherche du meilleur prédicteur temporel est effectué uniquement par l'encodeur, c'est donc un processus non-normatif (cf. 1.3.5). L'estimation de mouvement correspond à un algorithme de mise en correspondance de blocs ou "Block Matching", avec comme critère de sélection la minimisation du critère débit-distorsion. Cependant, l'estimation de mouvement n'étant pas normative, le choix du critère de sélection de l'algorithme dépend de chaque implémentation.

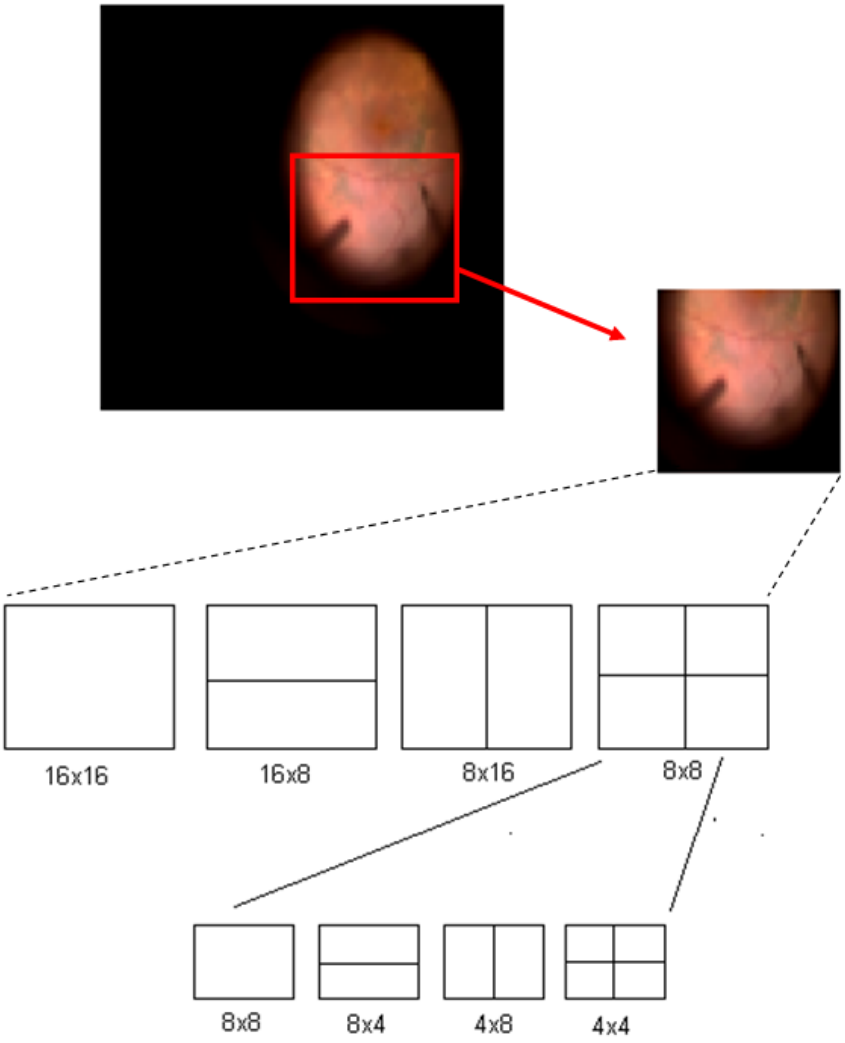


Figure 2.4 — Les modes de prédiction pour un bloc sélectionné

2.3.2.1 Codage de l'information de mouvement

En compression vidéo, afin d'exploiter au mieux les redondances temporelles entre les images successives, on introduit une notion de mouvement pour caractériser les différences entre ces images. Il existe différentes manières de coder cette information de mouvement. La manière la plus intuitive est sans doute d'affecter un vecteur déplacement à chaque objet d'une séquence vidéo. Ces méthodes ont été largement étudiées et intégrées dans la norme MPEG-4. Dans ce type de codage, chaque élément d'une scène est codé indépendamment : on sépare le fond des objets en mouvement. Dans chaque image, un objet est représenté par la déformation de son contour et par un vecteur mouvement traduisant son déplacement. Cependant ces méthodes n'ont pas été utilisées par les industriels malgré leur intégration dans la norme MPEG-4 car la segmentation de la scène en objets est difficile à obtenir avec des algorithmes rapides. De plus, le fond d'une séquence est rarement statique à cause des défauts des capteurs, et le débit du codage des contours et de leurs déformations n'est pas négligeable.

La représentation par bloc du mouvement est la plus utilisée dans les standards vidéo.

Ce partitionnement des images implique le codage d'un vecteur mouvement en coordonnées cartésiennes pour chaque bloc. Les algorithmes, à taille de blocs fixe, ne nécessitent pas de codage de la segmentation du mouvement et sont adaptés à la transformée de blocs de pixels carrés utilisée dans la quasitotalité des standards d'image fixe et vidéo : la DCT 2D. Le défaut de cette méthode est qu'elle n'exploite pas les corrélations spatiales du champ de vecteurs à l'intérieur des objets. La norme H.264/AVC applique une segmentation du macrobloc allant du 16x16 jusqu'au 4x4, où chaque bloc a un vecteur de mouvement propre.

Il existe deux types de codage de vecteur de mouvement :

- **Codage des vecteurs mouvement avec pertes :** dans les standards de codage d'image et de vidéo avec pertes, on quantifie uniquement l'information de texture afin de réduire l'entropie de cette information. Le codage des vecteurs mouvement avec pertes [16–18] a pour philosophie de traiter le mouvement en tant qu'information quantifiable, comme l'information de texture. La proportion de l'information de mouvement peut être plus élevée que celle liée à la texture, notamment à bas et trés bas débit. Ce type de codage a été testé dans H.264/AVC en boucle fermée. Pour chaque bloc, un pas de quantification du vecteur est sélectionné à l'aide du critère débit-distorsion (cf. 1.3.5). L'équation prend en compte le codage prédictif des vecteurs utilisé dans la norme H.264/AVC. Par conséquent, le prédicteur du vecteur est lui aussi quantifié. Cette méthode nécessite la transmission d'un pas de quantification pour chaque vecteur, ce qui engendre une nouvelle information de codage trés coûteuse. En pratique, la sélection du pas de quantification est effectuée image par image.
- **Codage prédictif des vecteurs mouvement :** le coût de l'information de mouvement, pour les algorithmes de compression utilisant une compensation de mouvement par bloc, dépend de trois paramètres :
  - la taille des blocs utilisés (plus les blocs sont de petite taille, plus le nombre de vecteurs à coder est élevé).
  - la résolution sous-pixellique utilisée pour la compensation de mouvement (la valeur d'un vecteur au 1/4 de pixel est multipliée par quatre).
  - l'entropie de l'information.

  Pour réduire l'entropie de l'information, on utilise généralement un codage prédictif. Dans ce codage on considère non plus l'entropie de la source mais celle des résidus de cette source. Pour le codage des vecteurs mouvement le résidu (cf. 1.3.8) va être transmis à la place du vecteur de mouvement.

  L'efficacité de la méthode dépend de la pertinence du prédicteur utilisé. Dans le cas des algorithmes hiérarchiques [19, 20], chaque vecteur mouvement peut être prédit par la valeur du vecteur parent. Dans les standards vidéo, afin d'exploiter les redondances spatiales des champs de vecteurs mouvement, la valeur du prédicteur dépend de la valeur des vecteurs voisins déja encodés/décodés. Ce prédicteur correspond à un médian (cf. 1.3.4). Dans [21] le prédicteur est le module de ces vecteurs. Dans [22] le prédicteur est le vecteur qui a le plus d'occurrences dans une fenêtre autour du vecteur à prédire. Les redondances temporelles entre les champs de vecteurs mouvement de deux images ont aussi été exploitées [23, 24].

2.3.2.2 Codage de l'information de mouvement dans H.264/AVC

La norme H.264/AVC utilise un codage prédictif des vecteurs mouvement. Le prédicteur $\hat{p}$ (cf. 1.3.8) est un médian spatial pour chacune des composantes (horizontale et verticale). Nous noterons ce prédicteur mv_{H264}. Les trois vecteurs voisins du vecteur courant mv, utilisés

pour le calcul du médian sont mv_a, mv_b, mv_c représentés dans la figure (2.5) . En fonction de la taille des blocs voisins et du bloc courant, le vecteur mv_c peut être remplacé par le vecteur mv_d. Pour des cas particuliers, dépendant des caractéristiques des blocs voisins, le prédicteur $\hat{p}$ peut être égal à mv_a ou mv_b ou mv_c.

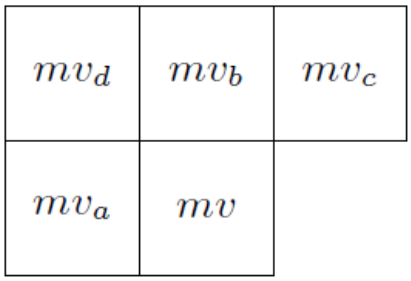


Figure 2.5 — Les modes de prédiction pour un bloc sélectionné

Ces caractéristiques sont l'appartenance des blocs à l'image, la taille du bloc courant et des blocs voisins et les images de référence utilisées pour le codage des blocs voisins. Par exemple, si un seul des vecteurs voisins a la même image de référence que le vecteur courant, la valeur de $\hat{p}$ est égale à ce vecteur voisin. De même, si l'un des blocs a, b, c, ou d est codé en Intra (modes pour lesquels il n'y a pas de vecteur mouvement) le vecteur mouvement pour ce bloc est égal à 0.

Les images B (cf. 1.3.1) utilisent des images dans le futur et le passé pour la compensation de mouvement [25]. De plus, il est possible d'utiliser des prédictions bidirectionnelles qui sont des combinaisons linéaires de deux compensations de mouvement impliquant l'utilisation de deux vecteurs mouvement (un vecteur par prédicteur de bloc). Les images de référence pour une prédiction bidirectionnelle peuvent se trouver dans le futur et le passé ou toutes dans le futur ou toutes dans le passé. Pour le médian, ceci ajoute une contrainte supplémentaire de direction (future ou passée ou les deux) pour les vecteurs voisins.

2.3.3 Prédiction Intra

La norme H.264/AVC utilise quatre prédicteurs pour le partitionnement 16x16 et neuf prédicteurs pour les partitionnements 4x4 et 8x8. Pour chaque bloc courant (4x4, 8x8 ou 16x16), la prédiction est calculée avec la ligne supérieure et la colonne de gauche qui appartiennent à des blocs déjà décodés (voir figure 2.6 et 2.7). Pour le mode 16x16, le prédicteur vertical est une copie de la ligne au-dessus du macrobloc pour chacune des lignes du prédicteur. De même le prédicteur horizontal est une copie de la colonne de gauche dans chacune des colonnes du prédicteur. Le troisième prédicteur est la moyenne de la ligne du haut et de la colonne de gauche. Ce prédicteur correspond à une prédiction, dans le domaine transformé, du premier coefficient du bloc. Enfin le prédicteur plan est la moyenne, pixel à pixel (dans le sens diagonal), de la ligne du haut et la colonne de gauche. Les neufs prédicteurs Intra, représentés dans la figure 2.7, sont le prédicteur DC (coefficient DC) et huit autres prédicteurs correspondant à huit directions, incluant les directions verticale et horizontale. Les prédicteurs pour les blocs de taille 8x8 sont construits de la même manière. Les indices des prédicteurs Intra 4x4 et 8x8 subissent eux aussi une forme de prédiction. En effet, si le prédicteur est égal au prédicteur le plus probable (calculé en fonction des deux blocs voisins en haut et à gauche), un seul bit sera transmis au décodeur dans le cas du CAVLC (Context-

adaptive variable-length coding) ou un seul bit sera utilisé pour la binarisation dans le cas du CABAC (Context-adaptive binary arithmetic coding). Si le prédicteur n'est pas égal à ce prédicteur le plus probable, la valeur du prédicteur sera codée ou binarisée avec 4 bits.

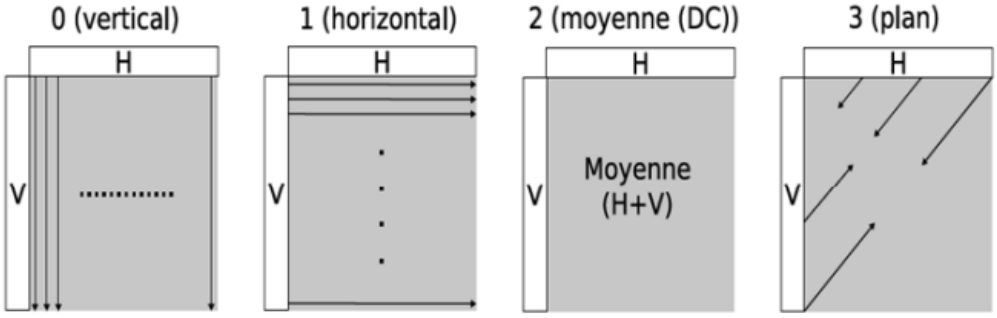


Figure 2.6 — Les quatre formes de prédiction (voir flèches) des blocs Intra 16x16 de la norme H.264/AVC

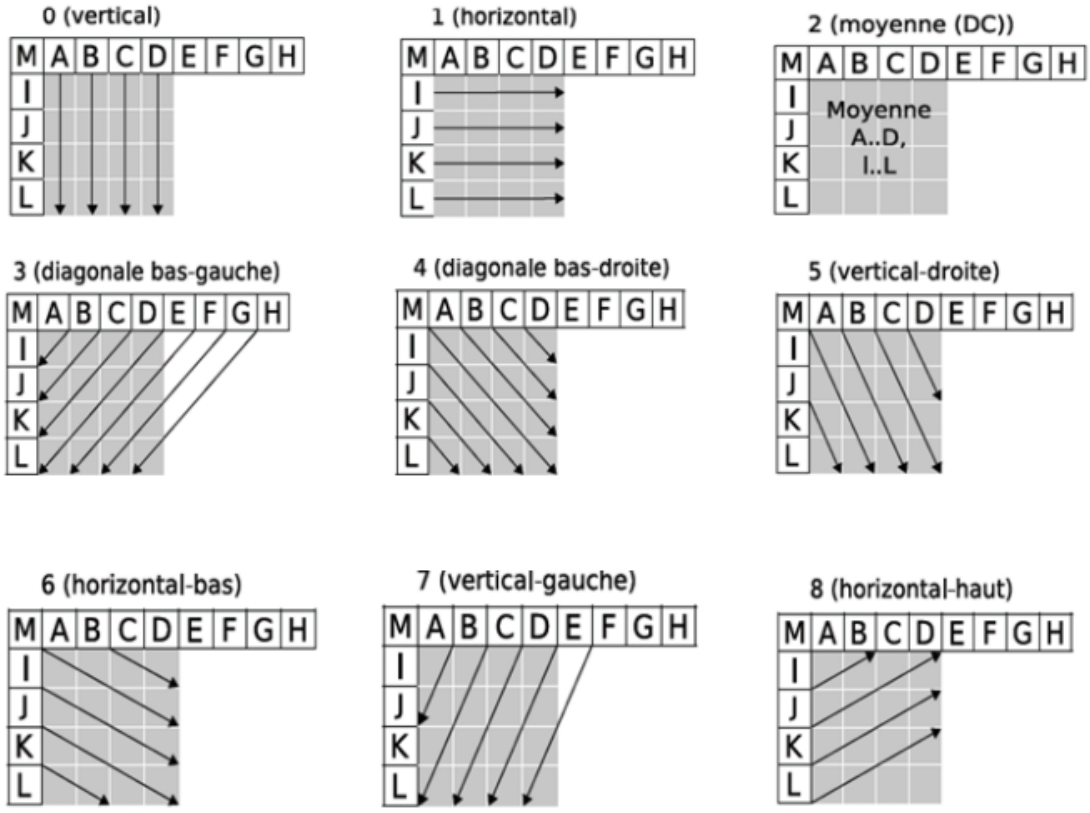


Figure 2.7 — Les neuf formes de prédiction (voir flèches) des blocs Intra 4x4 de la norme H.264/AVC

2.3.4 Métriques de distorsion et critère d'optimisation des modes

Le processus de sélection des modes n'est pas normatif dans H.264 / AVC, ce qui signifie que l'on peut choisir n'importe quel algorithme de sélection. Pour un macrobloc donné appar-

tenant à une image P ou B, on commence par déterminer les prédictions intra et inter images, correspondant à toutes les partitions de blocs possibles. Si l'image courante est une image I, seule la prédiction intra-image sera activée. Par exemple, on peut décider de générer toutes les prédictions inter et intra, pour l'ensemble des partitions définies et ne choisir qu'ensuite le meilleur mode pour le macrobloc. La sélection du meilleur mode se fait en général par le biais de la minimisation d'une fonction de coût. Ou peut aussi procèder par étape en choisissant le meilleur mode pour une partition donnée et pour un type de prédiction (intra ou inter). Parmi ces meilleurs modes, on sélectionne le meilleur pour le macrobloc. Cette technique est donc nécessairement sous optimale par rapport à la précédente qui met en concurrence tous les modes, en une seule fois. La figure 2.8 présente un exemple d'algorithme de sélection des modes. La fonction de coût qui est minimisée pour opérer la sélection peut être définie selon plusieurs critères prenant en compte plus ou moins de paramètres. Nous présentons dans cette section les critères basés uniquement sur une mesure de distorsion et ceux, plus complexes, basés sur une optimisation débit/distorsion.

2.3.5 Mesure de distorsion

Le moyen le plus simple pour évaluer la qualité d'une prédiction est de mesurer la distorsion D entre un bloc source et sa prédiction. Cette mesure évalue l'écart entre deux images en sommant pixel à pixel les erreurs faites. Le critère historiquement utilisé pour évaluer deux images entre elles, est le PSNR pour Peak Signal to Noise Ratio. Cette mesure de distorsion est donnée par la relation suivante :

$$PSNR = 10log_{10}(\frac{d^2}{EQM}) \tag{2.1}$$

Où d est la dynamique du signal et l'EQM est l'Erreur Quadratique Moyenne. Elle est définie de la façon suivante pour deux images x et y, de taille $(M \times N)$:

$$EQM = \frac{1}{MN}\sum_{i,j}(x(i,j) - y(i,j))^2 \tag{2.2}$$

Dans les critères d'évaluation, on utilise plus simplement, la somme des erreurs au carré ou SSE pour Sum Square of Errors :

$$SSE = \sum_{i,j}(x(i,j) - y(i,j))^2 \tag{2.3}$$

L'élévation au carré rend ces mesures particulièrement sensibles si les deux images sont trop éloignées. Un faible nombre de pixels ayant des valeurs trop différentes, suffit à perturber la mesure. Néanmoins, cette mesure de distorsion reste très efficace si la différence entre x et y correspond à un bruit gaussien, i.e. si les images ne présentent pas de différences majeures. Si ce n'est pas le cas, on préférera utiliser la SAD (Sum of Absolute Difference). Cette métrique consiste à calculer la norme l1 de l'image d'erreur :

$$SSE = \sum_{i,j} \mid (x(i,j) - y(i,j)) \mid \tag{2.4}$$

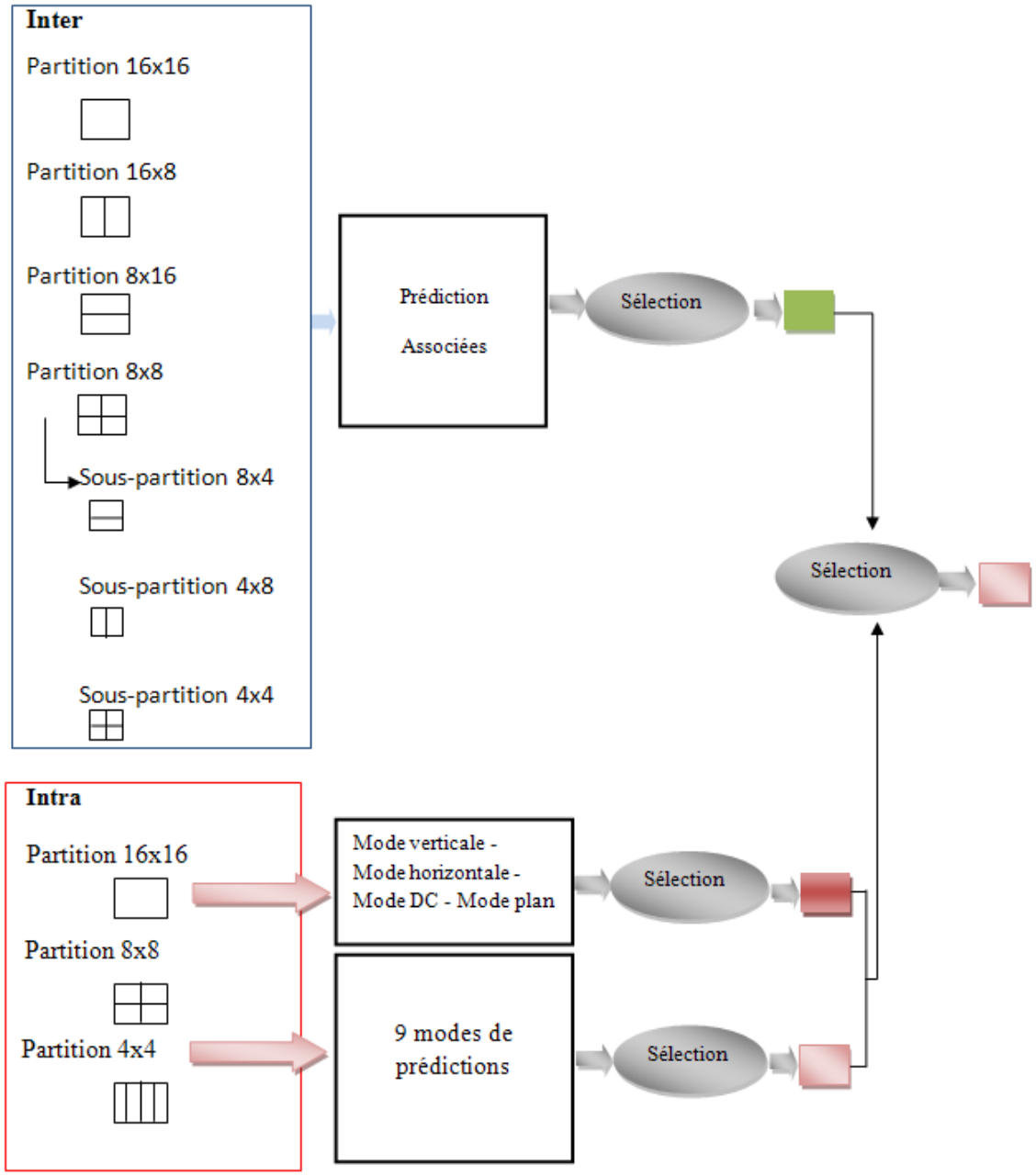


Figure 2.8 — Exemple de sélection des modes de prédiction

La SAD sera plus adaptée dans les cas où les deux images sont structurellement bien différentes.

2.3.6 Optimisation débit/distorsion

Pour affiner l'évaluation des méthodes de compression d'images, il a été défini un processus cherchant le meilleur compromis entre la qualité de la reconstruction et le nombre de bits nécessaires pour coder puis transmettre l'information résiduelle. Il s'agit d'un problème d'optimisation connu sous le nom RDO pour Rate Distorsion Optimization. Ce critère est basé sur une fonction de coût lagrangienne :

$$J_\lambda = D + \lambda.R \tag{2.5}$$

Où D est une mesure de distorsion, R représente le débit en bits et λ un paramètre d'ajustement qui dépend des contraintes de quantification. La métrique de distorsion utilisée est en général la SSE ou la SAD. Notons qu'il existe des décisions a priori et a posteriori. Dans le cadre de la décision a posteriori, la valeur exacte du débit R doit être connue, ce qui nécessite alors de passer par le processus complet de codage (transformation et quantification). La SSE quant à elle impose la reconstruction via les étapes de quantification et transformation inverses. La technique est performante mais reste coûteuse car cela nécessite de répéter le processus pour chaque bloc et pour chaque mode. L'alternative est l'utilisation d'une approche moins complexe mais néanmoins sous-optimale, qui consiste à évaluer a priori le coût qu'aurait le bloc en fin de processus. Pour estimer le débit R, on se base sur des modèles empiriques d'optimisation débit/distorsion, permettant d'approcher les performances des algorithmes a posteriori.

2.3.7 Calcul du résidu

Le résidu du bloc courant, appelé aussi erreur de prédiction, est la différence entre un prédicteur et ce bloc courant. L'expression du résidu est donnée dans l'équation (1.6) où p(x,y) est le pixel du bloc courant à la position (x,y) et $\hat{p}$ est le prédicteur :

$$e(x, y) = p(x, y) - \hat{p}(x, y) \tag{2.6}$$

L'opération de prédiction inverse est donnée dans l'équation (1.7). Le prédicteur $\hat{p}$ est ajouté au résidu pour retrouver les pixels p(x,y) du bloc courant, le décodeur calcule le prédicteur et $\hat{p}$ à partir des informations déjà décodées et extraites du flux binaire :

$$p(x, y) = \hat{p}(x, y) + e(x, y) \tag{2.7}$$

2.3.8 Transformation

Dans le même contexte que les normes précédentes, la phase de transformation-quantication est appliquée dans le but de coder le signal d'erreur de prédiction. La tâche de la transformée consiste à réduire les redondances spatiales du signal d'erreur. Toutes les anciennes normes tels que le MPEG-1 et MPEG-2 appliquaient une DCT de taille (8×8) sur chaque bloc de l'image. En revanche, la norme H.264/AVC utilise une transformée entière (a les mêmes propriétés que la DCT classique [26]). La matrice de transformation est généralement composé de (4×4) éléments, mais peut être réduite à (2×2) éléments pour le codage de certaines informations de chrominance [11]. La diminution de la taille de la fenêtre d'analyse permet à l'encodeur de mieux adapter le codage de l'erreur de prédiction aux frontières des objets mouvants. En effet, la taille du bloc est similaire aux dimensions de la plus petite zone d'analyse de l'estimation Inter ou Intra ((4×4) pixels) et la transformée s'ajuste donc mieux aux erreurs de prédiction locales.

Il existe trois types différents de transformées. Le premier est appliqué à tous les échantillons quel que soit le mode de prédiction utilisé, cette matrice de transformation H_1, est composée de 4x4 éléments, sa structure est exposée dans l'équation suivante :

$$H_1 = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{pmatrix}$$

Si le macrobloc est prédit en utilisant le mode Intra (16×16), la seconde transformée est appliquée en plus de la première. Cette dernière convertit les seize coefficients DC correspondant à l'intensité moyenne des blocs transformés d'un macrobloc en utilisant une transformée de Hadamard dont la taille est de (4×4) composantes.

$$H_2 = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix}$$

Le troisième type se rapporte aussi à une transformée de Hadamard mais de taille (2×2). Elle est utilisée pour le codage des quatre coefficients DC contenus dans un macrobloc de chrominance.

$$H_3 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

L'opération de transformation dans le codec H.264/AVC se traduit par l'équation suivante :

$$Y = H_i.X.H_i^T. \tag{2.8}$$

Ou Y est la matrice transformée, X le signal d'entrée et H_i peut représenter $H_1(1,2)$, $H_2(1,3)$, $H_1(1,4)$

En comparaison avec la DCT, les matrices de transformation du H.264/AVC sont composées seulement de nombres entiers dans un intervalle compris entre -2 et +2. Ce principe permet de calculer la transformée et son inverse sur seize bits en utilisant seulement des opérations d'addition et de soustraction. Dans le cas d'une projection de Hadamard, seules l'addition et la soustraction sont nécessaires. De plus, les disparités liées aux approximations du calcul flottant sont complètement évitées grâce à l'utilisation exclusive d'opérations sur des entiers. Tous les coefficients sont ensuite quantifiés par le biais d'un quantificateur scalaire.

2.3.9 Quantification

La quantification scalaire a pour but de réduire l'espace des valeurs des transformées pour réduire l'entropie du signal. Cette opération Q consiste à diviser chaque coefficient du bloc transformé par son coefficient de quantication provenant d'une matrice de quantification et à ne garder que la partie entière. La taille du pas de quantication est choisie par un paramètre QP qui peut prendre cinquante deux valeurs possibles. La taille du pas double lorsque la

variable QP est incrémentée de 6. Une augmentation de QP de 1 entraîne un accroissement du débit des données d'environ 12.5 %

L'opération conjointe de quantification et de dé-quantification est donnée par la formule suivante

$$Q^{-1}Q^{(x)} \longmapsto mqsix \in]mq - \frac{q}{2}, mq - \frac{q}{2}[\tag{2.9}$$

Où x est un coefficient transformé et q est le coefficient ayant la même position dans la matrice de quantification. Pour que la reconstruction du signal soit optimale, un "offset" de quantification est ajouté à chaque coefficient. Dans le logiciel de référence de la norme H.264/AVC [11], cet offset de quantification est fixé de manière adaptative en fonction de statistiques obtenues sur les blocs précédemment quantifiés.

2.3.10 Codage entropique

Le H.264/AVC propose deux méthodes alternatives de codage entropique : une technique de faible complexité basée sur l'usage de contextes adaptatifs contenant des mots de code VLC, nommé CAVLC (Context-based Adaptive Variable Length Coding) [26], et un algorithme plus coûteux fondé sur un codage arithmétique reposant sur des tables évolutives, le CABAC (Context-based Adaptive Binary Arithmetic Coding) [27]. Les deux méthodes représentent des améliorations majeures en terme d'efficacité de compression en comparaison avec les techniques de codage statistique traditionnelles. Dans les anciennes normes, l'encodage de chaque élément de syntaxe était basé sur des tables VLC fixées (une distribution de probabilité était associé chaque élément). Cependant, des études pratiques ont rapidement démontré que les signaux étaient rarement stationnaires et que l'utilisation de tables adaptatives (contextuelles) était plus efficace pour comprimer les données. Des modèles contextuels ont donc été intégrés dans le processus d'encodage entropique.

Le codage CAVLC est le plus utilisé, il fournit un codage efficace de faible complexité est inclu dans tous les profils définis par la norme H.264. Dans le codage CAVLC, deux techniques de compression sont utilisées. La première, basée sur un codage Exponential-Golomb (noté Exp-Golomb dans la suite) [28], se charge d'encoder tous les paramètres de codage (type de macrobloc, pas de quantication, vecteurs de mouvement, etc) à l'exception des résidus de prédiction. Ces résidus sont encodés par la deuxième méthode, plus compliquée, mais permettant de comprimer les données de manière adaptative.

Dans le codage CAVLC, les résidus de prédiction (transformés et quantifiés) sont encodés de manière indépendante en suivant une procédure spécifique, les coefficients du bloc transformé sont codés en se basant sur un schéma de balayage spécifique (zig-zag scan). On trouvera plus de détails dans [14].

2.3.11 Le filtre anti-blocs

Une caractéristique particulière du codage par blocs correspond à la production accidentelle d'artefacts entre des ensembles successifs. Le filtrage du bord des blocs représente un outil puissant qui permet de réduire considérablement la visibilité du découpage en blocs [29]. Dans le principe, le lissage peut être considéré comme un traitement final, se rapportant seulement aux images qui sont affichées. Une plus haute qualité visuelle peut encore être atteinte en

Valeur positif	Valeur Signée	Mot de code
0	0	1
1	+1	010
3	-1	011
4	+2	00100
5	-2	00101
6	+3	00111
7	-3	0001000
8	+4	0001001
9	-4	0001010
10	+5	0001001
...	...	...

Figure 2.9 — Table de correspondance du codage Exp-Golomb

intégrant le filtre dans la boucle d'encodage. En effet, toutes les images de référence passées, utilisées pour la compensation en mouvement, seront des versions corrigées des images reconstruites. C'est pour cette raison, que le H.264/AVC incorpore cette technologie dans la boucle de traitement. Ce filtre est hautement adaptatif car il dépend de plusieurs éléments de syntaxe mais aussi des caractéristiques locales de l'image. Ces différentes contraintes permettent de contrôler la souplesse du traitement de filtrage.

La figure 2.10 illustre le principe de l'opération en utilisant une représentation d'un bloc (4×4) blocs en une dimension. Avant de réaliser le traitement, des contraintes doivent être respectées.

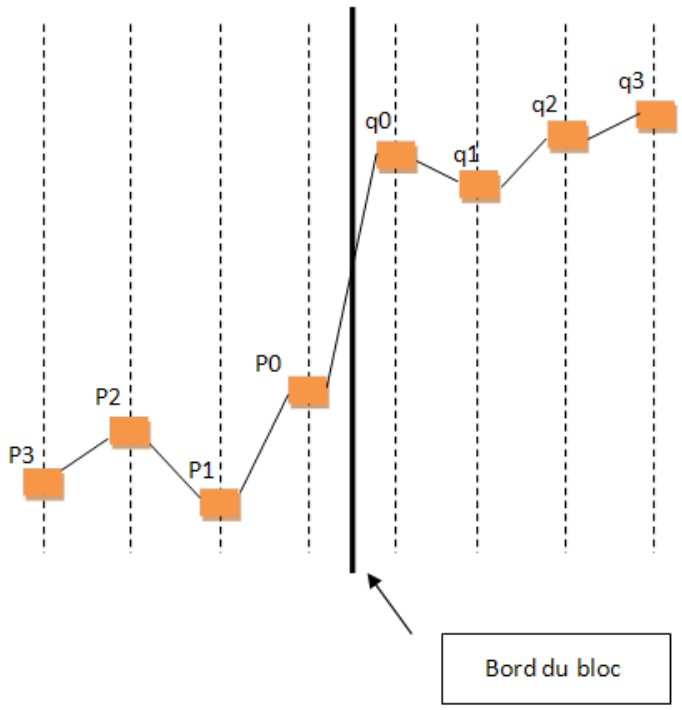


Figure 2.10 — Shéma du filtre anti-blocs

Ces contraintes dépendent de la valeur des échantillons et du paramètre de quantification QP. Ainsi, le filtrage de p0 et q0 intervient seulement si chacune des expressions suivantes est vérifiée :

1. $| p_0 - q_0 | < \beta(QP)$
2. $| p_1 - p_0 | < \alpha(QP)$
3. $| q_1 - q_0 | < \alpha(QP)$

Où le seuil $\alpha(QP)$ est considéré plus faible que $\beta(QP)$.

L'idée de base repose sur une simple constatation. Si la différence entre les échantillons proches du bord d'un bloc est relativement importante, il est probable que ce phénomène corresponde à un artefact de bloc et doive donc être réduit. En revanche, en cas d'écart trop élevé, ne pouvant donc pas être expliqué par la quantification, le signal représente plutôt l'information propre de l'image source. Dans ce dernier cas, le lissage n'est pas appliqué au bloc.

2.4 Améliorations apportées par rapport aux autres encodeurs

Jusqu'à l'apparition d'H.264 AVC, la transformée ne s'opérait que sur des blocs de taille (8×8) ne permettant pas une décorrélation fine du signal. Il a donc été introduit une transformation fréquentielle sur des blocs de taille (4×4). L'exploitation des corrélations spatiales résiduelles à une résolution plus fine permet d'améliorer la représentation des détails. On peut remarquer que la transformation utilisée est définie de manière exacte (précision entière) afin d'éviter les erreurs d'arrondis. Aux modifications présentées précédemment, s'ajoutent encore quelques détails importants que nous présentons ci-dessous.

La nouvelle norme permet de fournir jusqu'à seize vecteurs de mouvement par macrobloc. Jusqu'alors, seuls deux voire quatre vecteurs de mouvement étaient définis, ce qui limitait les performances de la prédiction temporelle.

- La nouvelle norme permet de fournir jusqu'à seize vecteurs de mouvement par macrobloc. Jusqu'alors, seuls deux voire quatre vecteurs de mouvement étaient définis, ce qui limitait les performances de la prédiction temporelle.
- De même, le passage à une précision supérieure dans le calcul des vecteurs de mouvement a été une avancée majeure. En effet, par interpolation d'image, la norme H.264/AVC autorise une recherche au quart de pixel améliorant ainsi considérablement la précision des vecteurs de mouvement.
- Une autre spécificité d'H.264/AVC a été l'introduction de modes supplémentaires, nommés modes directs, dont le but est de déduire les vecteurs de mouvement. Le principe consiste à éviter de les calculer en les estimant à partir des vecteurs définis pour les blocs voisins. Il existe deux modes directs : l'un spatial, l'autre temporel. En spatial, les vecteurs voisins correspondent à ceux retenus pour les blocs limitrophes au bloc courant. Le mode direct temporel utilise quant à lui, l'information du bloc colocalisé dans l'image de référence. Ces nouveaux modes sont appliqués pour les images B. Il existe également un mode direct spatial pour les images P, pour la prédiction inter (16×16). On gagne ainsi en débit puisque l'on évite d'avoir à coder toute l'information relative aux vecteurs de mouvement.

– La quantification a elle aussi été légèrement modifiée. On a augmenté le nombre de pas de quantification jusqu'à 52 niveaux afin d'améliorer la représentation du signal.

2.5 Conclusion

Dans ce chapitre, nous avons décrit les principes de base de construction d'un codeur vidéo afin de présenter les étapes de la chaîne de codage. Nous avons comparé ensuite plusieurs possibilités de codages et présenté les différentes étapes du codeur H.264/AVC : la décorrélation temporelle, la décorrélation spatiale, la quantification et le codeur entropique.

Le fondement de ce codage réside dans le fait qu'une seule technique de codage ne peut pas être efficace pour toutes les zones de l'image. Il est donc essentiel d'étudier plusieurs possibilités de codage pour sélectionner celle qui obtient la meilleure efficacité. La compétition entre les différentes techniques se fait à plusieurs niveaux : au niveau séquence pour les choix applicatifs, au niveau image pour sélectionner le type d'image, au niveau objet pour le codage dynamique et au niveau bloc pour le codage de même nom. La sélection parmi l'ensemble des possibilités de codage se fait par le critère débit-distorsion. Ce critère pondère le débit en fonction de la distorsion avec le paramètre de Lagrange. Ce paramètre dépend de la quantification, du type de codage mis en compétition et des applications visées. La sélection des possibilités de codage peut se faire par des choix sous-optimaux basés sur des a priori qui sont généralement utilisés pour réduire la complexité de calcul.

Ces techniques, ainsi que plusieurs autres, aident H.264/AVC à dépasser significativement les standards précédents, dans une grande variété de circonstances et dans une grande variété d'environnements applicatifs. L'objectif est de refléter au mieux l'appréciation subjective de la qualité de compression provenant d'une multitude de possibilités de prédictions.

Comme nous l'avons rappelé dans le chapitre 1, l'idée fondamentale de la CBVR est de décrire d'une manière compacte une vidéo par une signature numérique et le plus rapidement possible dans le cas de notre étude (mode peropératoire), puis apparier la requête aux vidéos les plus ressemblantes dans la base de données du point de vue similitude de leurs signatures. Dans le cadre de cette thèse, nous avons voulu explorer les possibilités de créer ces signatures en utilisant les informations utilisées pour compresser les vidéos. Il s'agissait de profiter des différentes méthodes de compression et de leur différentes étapes pour extraire l'information pertinente qui permet de caractériser les vidéos. Dans ce chapitre, nous avons présenté de manière globale les principaux concepts de la compression de vidéo, en approfondissant ceux que nous utilisons dans notre travail de recherche. Dans les prochains chapitres, nous allons expliciter comment nous avons construit des signatures, en utilisant principalement les étapes de prédiction, transformation et quantification de l'architecture générale du processus de compression. Elles nous permettront de rester dans le cadre de l'utilisation de paramètres de bas niveaux caractérisant le mouvement, la texture et la couleur.

Bibliographie

[1] C. Reader. History of MPEG Video Compression - Ver. 4.0. Joint Video Team (JVT) doc, 2002. JVT-E066.

[2] ITU, CCITT. Recommandation IT-81, Information Technology - Digital Compression and Coding of Continuous-Tone Still Images - Requirements and Guidelines (JPEG), 1992.

[3] R. D. Kell. Improvements relating to electric picture transmission systems. Technical report, British patent No. 341.811, 1929.

[4] ITU-T. Codec for videoconferencing using primary digital group management. Technical report, ITU-T Rec. H.120, version 1, 1984.

[5] A. Habibi. Hybrid coding of pictorial data. IEEE Trans. on Communications, 22(5) :614 624, 1974.

[6] ITU-T. Video codec for audiovisual services at px64 kbits/s. Technical report, ITU-T Rec. H.261, version 1, nov. 1990.

[7] ISO/IEC JTC 1. Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbis/s - part 2 : Video. Technical report, ISO/IEC 11172-2 (MPEG-1), mar. 1993.

[8] ISO/IEC JTC 1/SC 29. Generic coding of moving pictures and associated audio information : Systems. Technical report, ISO/IEC 13818-1 (MPEG-2 Part 1), 1996.

[9] ITU-T. Video coding for low bit rate communication. Technical report, ITU-T Rec. H.263, version 1, nov. 1995.

[10] ISO/IEC JTC 1. Coding of audio-visual objects - part 2 : Video. Technical report, ISO/IEC 14496-2 (MPEG-4 visual version 1), apr. 1999.

[11] ITU-T and ISO/IEC JTC 1. Advanced video coding for generic audiovisual services. Technical report, ITU-T Rec. H.264, and ISO/IEC 14496-10 AVC, nov. 2003.

[12] K. Sayood. Introduction to Data Compression, Second Edition. Morgan Kaufmann, San Francisco, 2000.

[13] A. K. Jain. Fundamentals of Digital Image Processing. Prentice-Hall, Englewood Cliffs, NJ, 1989.

[14] Advanced Video Coding for Generic Audiovisual Services, ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG-4 AVC). ITU-T and ISO/IEC JTC 1, Version 1 : Mai 2003, Version 2 : Mai 2004, Version 3 : Mars 2005, Version 4 : Septembre. 2005, Version 5 and Version 6 : Juin 2006, Version 7 : Avril. 2007.

[15] I. Richardson. H.264 and MPEG-4 Video Compression : Video Coding for Next-Generation Multimedia. John Wiley and Sons, 2003.

[16] R.L. Joshi, T.R. Fischer, et R.H. Bamberger. Lossy encoding of motion vectors using entropy-constrained vector quantization. 3 :109112, Octobre 1995.

[17] L. Yoon Yung et J.W. Woods. Motion vector quantization for video coding. Image Processing, IEEE Transactions on, 4(3) :378382, Mars 1995.

[18] A.L. Da Silva Cruz et J.W. Woods. Adaptive motion vector quantization for video coding. In Image Processing, ICIP, IEEE International Conference on, volume 2, pages 867870, Vancouver, Canada, Octobre 2000.

[19] A. Deever et S. Hemami. Dense motion field reduction for motion estimation. In Signals, Syst. and Comput., volume 2, pages 944948, Novembre 1998.

[20] M.H. Chan, B.Y. Yu, et A.G. Constantinides. Variable size block matching motion compensation with applications to video coding. Proc. Inst. Elec. Eng., 137 :205212, Août 1990.

[21] T. Ebrahimi. A new technique for motion field segmentation and coding for very low bitrate video coding applications. In Image Processing, ICIP, IEEE International Conference on, volume 2, pages 433437, Austin, Texas, USA, Novembre 1994.

[22] R. Krishnamurthy Compactly Encoded Optical Flow Fields for Motion Compensated Video Compression and Processing. Thèse de Doctorat, Rensselaer Polytechnic Institute, Troy, New York, USA, 1997.

[23] Y.Q. Zhang et S. Zafar. Predictive block-matching motion estimation for TV coding. II. Interframe prediction. Broadcasting, IEEE Transactions on, 37(3) :102105, Septembre 1991.

[24] J. Yeh, M. Vetterli, et M. Khansari. Motion compensation of motion vectors. In Image Processing, ICIP, IEEE International Conference on, volume 1, pages 574577, Washington, District de Columbia, USA, Octobre 1995.

[25] M. Flierl et B. Girod. Generalized B pictures and the draft H.264/AVC video compression standard. IEEE Trans. on Circuits and System for Video Technology, 13(7) :587597, Juillet 2003.

[26] G. Bjontegaard and K. Lillevold. Context-adaptive VLC coding of coefficients. Technical report, JVT, 2002.

[27] D. Marpe, H. Schwarz, and T. Weigand. Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard. IEEE Trans. on Circuits and Systems for Video Technology, 13(7) :620 636, 2003.

[28] D. Marpe, G. Blattermann, and T. Wiegand. Adaptive codes for H.26L. Technical report, JVT, 2001.

[29] P. List, A. Joch, J. Lainema, G. Bjøntegaard, and M. Karczewicz. Adaptive deblocking filter. IEEE Trans. on Circuits and Systems for Video Technology, 13(7) :614 619, 2003.

CHAPITRE 3

INDEXATION ET RECHERCHE DE VIDEO DANS LE DOMAINE COMPRESSÉ : MÉTHODES DÉVELOPPÉES

Nous avons déjà expliqué, dans le chapite I, l'intérêt de travailler dans le domaine compressé : toute l'information utile doit s'y trouver et cela nous permet de bénéficier potentiellement, avec les nouvelles normes, d'algorithmes de calcul des paramètres décrivant les vidéos beaucoup plus performants que d'autres méthodes d'extraction et de représentation de données vidéo. Les premiers systèmes d'indexation de vidéo par le contenu, comme ceux étudiés dans [1] ou [2], ont obtenu un certain succès dans la gestion de requêtes générales en utilisant des caractéristiques globales de la vidéo. Toutefois, ces systèmes ont leurs limites comme nous allons le voir au cours de ce chapitre. L'utilisation de caractéristiques globales ne tient pas compte des contraintes d'organisation spatiale de l'information. Ensuite elle ne reflète pas la manière dont nous percevons le contenu. Finalement elle ne permet pas de représenter efficacement le contenu sémantique de la scène. Comme nous avons pu le voir dans le chapitre 1, les méthodes d'indexation des régions sont peu répandues. La difficulté de la segmentation, la complexité de la représentation et des mesures de comparaison sont les principales barrières à leur développement. Cependant l'intérêt de travailler sur des régions est immense : outre le fait d'apporter une description qui est en accord avec notre système visuel, l'analyse des régions ouvre les portes à une étude plus approfondie du contenu comme la détection des objets/régions identifiés importants dans les scènes pour ensuite avoir une caractérisation plus efficace et moins coûteuse. Cette approche s'apparente plus au comportement que nous adoptons pour observer notre environnement. De plus elle répond mieux au besoin réel des utilisateurs qui recherchent des objets/régions présents dans des scènes dont la composition change. Nous avons donc porté notre effort sur la représentation efficace et compacte du contenu des vidéos en utilisant les régions, les travaux étant conduits dans le cadre de l'indexation de vidéos par le contenu visuel.

Nous présentons rapidement en début de chapitre le logiciel 'JM-reference' que nous util-

isons pour extraire les données contenues dans la norme H264. Ensuite, nous développons les trois méthodes que nous proposons pour représenter le contenu visuel des vidéos : la première méthode consiste à caractériser globalement la vidéo en utilisant des histogrammes de directions de mouvement. Les deux autres méthodes sont basées sur une segmentation spatio-temporelle et un suivi des régions dans la séquence vidéo, pour décrire le contenu des régions identifiées comme les plus importantes visuellement. Les approches que nous proposons produisent des signatures caractérisant la vidéo d'une manière synthétique et structurée ; elles sont de plus génériques car elles s'adaptent aux vidéos étudiées en exploitant les données issues du domaine de la compression pour la construction des signatures. Finalement, les méthodes de comparaison de signatures DTW (Dynamic Time Warping), EMD (Earth Mover's Distance) et une nouvelle approche de mesure de distance dite EFDTW (Extended Fast Dynamic Time Warping) sont présentées.

3.1 JM référence (Joint Model)

Dans notre travail, nous avons utilisé la norme de codage vidéo H.264 nommée aussi "MPEG-4 Advanced Video Coding". Cette norme comprend de nombreuses améliorations techniques qui lui permettent de compresser beaucoup plus efficacement les vidéos que les normes précédentes (H.261, MPEG1, MPEG2, MPEG4 part2/ASP) [3] (cf. chapitre 2) et elle est à ce jour très utilisée pour la compression vidéo grand public. Notre objectif était d'utiliser directement les informations déjà calculées, présentes dans le flux compressé MPEG4, pour limiter les temps de calcul. Pour extraire ces informations, nous avions le choix entre les encodeurs H.264 [4] et JM référence [5]. Si le premier est plus performant, le code source du second est plus simple d'accès.

Le JVT (Joint Video Team) a développé ce logiciel afin d'expérimenter et de valider les fonctionnalités de H264. Le logiciel JM (Joint Model) en est à sa version 18.4, et les sources sont librement téléchargeables sur la page web des logiciels du JVT [6]. En utilisant la norme MPEG-4 AVC et en analysant le code source de JM référence, nous avons localisé les fonctions importantes (l'estimation de mouvement, la décision d'encodage, les paramètres de décomposition des images en blocs (la taille des blocs), le mode de codage utilisé (inter ou intra), la taille des GOP (Group of pictures), etc), que nous utilisons pour extraire les signatures des vidéos par la suite. Lors de l'encodage, le logiciel fournit par ailleurs une ligne de statistiques utiles pour chaque image. La figure 3.1 est une capture d'écran de ce que fournit le logiciel pour chaque image, ce sont les statistiques globales de l'encodage de la séquence vidéo.

```
-------------------------------------------------------------------------------
Frame     Bit/pic    QP   SnrY    SnrU    SnrV    Time(ms) MET(ms) Frm/Fld Ref
-------------------------------------------------------------------------------
00000(NVB)     168
00000(IDR)   39624   28  35.389  40.278  40.426        89       0    FRM    1
00001( P )   15744   28  34.339  40.122  40.474       525     402    FRM    1
-------------------------------------------------------------------------------
 Total Frames:  2 (2)
 Leaky BucketRateFile does not have valid entries.
 Using rate calculated from avg. rate
 Number Leaky Buckets: 8
     Rmin     Bmin     Fmin
   207630    39624    39624
   259537    39624    39624
   311444    39624    39624
   363351    39624    39624
   415258    39624    39624
   467165    39624    39624
   519072    39624    39624
   570979    39624    39624
------------------ Average data all frames  ---------------------------------

 Total encoding time for the seq.  :   0.615 sec (3.25 fps)
 Total ME time for sequence        :   0.402 sec

 Y { PSNR (dB), cSNR (dB), MSE }   : { 34.86, 34.83, 21.37 }
 U { PSNR (dB), cSNR (dB), MSE }   : { 40.20, 40.20,  6.21 }
 V { PSNR (dB), cSNR (dB), MSE }   : { 40.45, 40.45,  5.86 }

 Total bits                        : 55536 (I 39624, P 15744, NVB 168)
 Bit rate (kbit/s)  @ 7.50 Hz     : 208.26
 Bits to avoid Startcode Emulation : 0
 Bits for parameter sets           : 168

-------------------------------------------------------------------------------
Exit JM18.3(FRExt) encoder ver 18.3
```

Figure 3.1 — Statistique globale de l'encodage

Définition des statistiques affichées :
- Type de l'image : (I pour Intra, P pour Predict, IDR pour la première image). Il existe d'autres types d'images comme la B qui est une image interpolée entre deux images P, moyenne des deux P sans refaire d'estimation de mouvement. (cf. chapitre 2)
- Bit/pic : le poids de l'image en bits.
- QP : le coefficient de quantification, de 1 (sans pertes) à 51 (forte compression).
- SNR : la mesure de rapport signal à bruit entre l'image originale et l'image codée, dans chaque composante Y, U, et V. Pour les mesures, il est recommandé de prendre en compte le SNR de la luminance de l'image.
- Time : le temps d'encodage de l'image.
- MET : le temps d'estimation de mouvement de l'image.
- Frm/Fld : le mode de codage de l'image.
- Ref : l'indicateur de référence courante de l'image.

3.2 Signatures basées sur l'orientation et l'intensité de mouvement

Notre première méthode consiste à définir une représentation globale du mouvement dans chaque image de la séquence, après l'extraction du mouvement en utilisant le logiciel JM Reference. Chaque image de la séquence est représentée par un histogramme basé sur la direction de ses vecteurs de mouvement (voir figure 3.2).

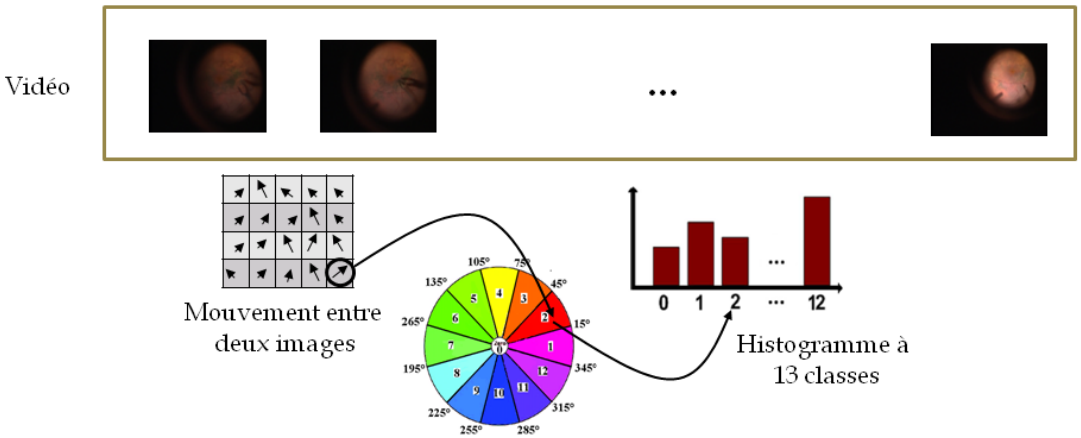


Figure 3.2 — Exemple de classification de vecteur de mouvements d'un macrobloc

La direction du vecteur V = (x, y) en un point (x, y) est calculée par la formule suivante :

$$W(V) = \begin{cases} arccos\frac{x}{|V|} & si \quad y \geq 0 \\ 2\Pi - arccos\frac{x}{|V|} & si \quad y < 0 \end{cases}$$

Où $W(V)$ représente la direction du mouvement, $\mid V \mid$: la norme euclidienne.

Nous construisons un histogramme à 13 classes de ces directions de vecteurs mouvement, La direction du mouvement $W(V)$ est affectée à l'une des 13 classes de l'histogramme (voir

figure 3.2) selon la formule suivante :

$$Hist(V) = \begin{cases} 0 & si \quad V = (0,0) \\ 1 + ([W(V)\frac{K}{2\Pi} + \frac{1}{2}] \mathrm{mod} K) & sinon \end{cases}$$

Nous avons choisi $K = 12$ pour avoir une bonne classification des vecteurs, nous avons donc 13 classes : 12 de direction, plus la classe contenant les déplacements nuls ($V = (0, 0)$).

La figure 3.2 montre un exemple de classification de vecteur de mouvement d'un macrobloc. Nous utilisons l'histogramme de l'image pour extraire la signature de l'image et ensuite celle de la vidéo.

Pour extraire la signature de l'image, le nombre de vecteurs de la classe dominante (contenant le plus de vecteurs, appelée Direction dominante), le numéro de la classe (Angle : la classe 2 dans l'exemple illustré par la figure 3.2), ainsi que l'intensité des vecteurs de mouvement appartenant à la classe dominante (voir formule suivante), sont extraits [7] :

$$Intensité_C = \frac{1}{C} \sum_{i=1}^{C} \mid V \mid \tag{3.1}$$

C représente le nombre de vecteurs de mouvement appartenant à la classe dominante.

Notre vecteur descripteur contient ainsi 3 paramètres par image : < Direction, Angle, Intensité >. Nous illustrons ci-dessous le processus d'extraction de signature pour une séquence vidéo de la base (voir la figure 3.3)

La signature d'une séquence vidéo obtenue par cette méthode donne des résultats intéressants comme nous le verrons dans le chapitre 4 (Résultats). Cependant, dans ce travail nous devons concevoir des approches avec un temps de calcul admissible dans un contexte peropératoire. La méthode proposée est lente malgré son efficacité. Pour pallier ce problème nous proposons dans le paragraphe suivant une autre approche : seules les images I (cf. chapitre 2) sont considérées ; de plus, nous travaillons sur des régions et non plus sur des blocs.

3.3 Signatures basées sur le suivi des régions homogènes entre images I

Dans cette méthode, nous nous intéressons au suivi de régions, le but étant d'extraire la trajectoire de chaque région segmentée (localisée) afin de caractériser la vidéo. Nous avons choisi la méthode de croissance de régions présentée dans [8] ("seeded region-growing"), comme méthode de segmentation et les filtres de Kalman pour la prédiction (suivi de régions).

Dans cette approche, nous n'utilisons que les images I issues du codeur MPEG-4. L'encodeur vidéo MPEG utilisé dans ce travail produit une image I toutes les 15 images (15 images correspondant à un GOP) (cf. chapitre 2). 1 seconde de vidéo contient 25 images, et par conséquent, comprend au moins une image I. Les images I contiennent les données les plus importantes dans la vidéo, elles se retrouvent au moins à chaque changement de plan. Pour réduire le temps de calcul de la signature, tout en gardant le contenu principal d'une

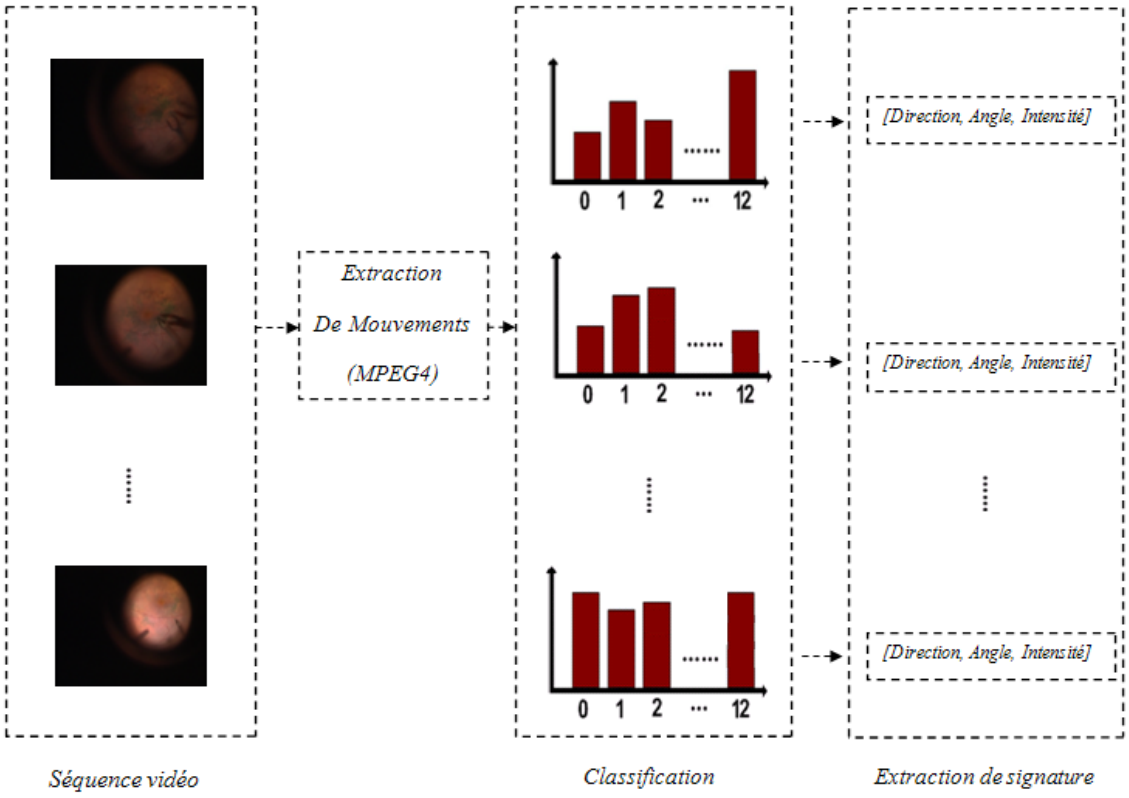


Figure 3.3 — Etapes de la caractérisation de la vidéo

scène, nous nous concentrons sur les informations de macroblocs dans les images I. Les informations de mouvement, de texture (résidus) sont extraites de chaque paire (I-image, P-image suivante) (voir la figure 3.4).

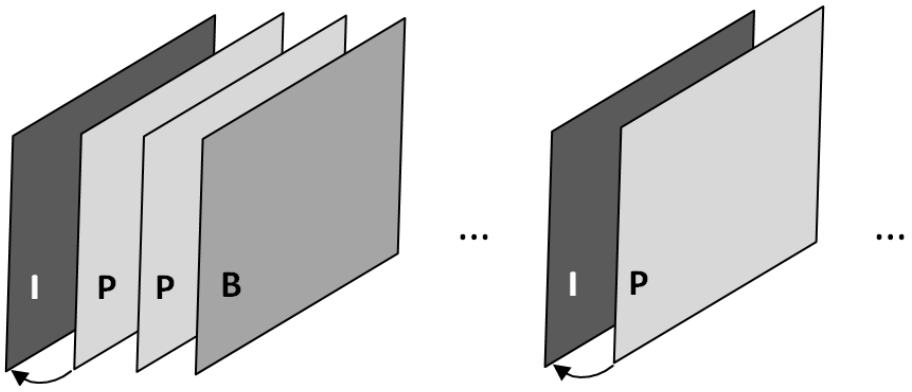


Figure 3.4 — Paires (I-image , P-image)

3.3.1 Segmentation par croissance de région à partir d'un germe (seeded-region growing)

Chaque image I de la séquence est segmentée en régions en se basant sur le mouvement extrait depuis le standard MPEG.

1. L'algorithme consiste à chercher un ensemble de 5 blocs et le définir comme étant le germe (voir étape 1 de l'algorithme). Puis faire croître le germe en une région en prenant les blocs environnants qui satisfont certains critères. Cette étape est répétée jusqu'à ce qu'il ne reste plus de bloc candidat dans l'image pour former le germe. Nous obtenons alors un ensemble de régions.
2. Les régions présentant des caractéristiques voisines sont fusionnées selon des critères d'homogénéisation de vecteurs de mouvement.

Les étapes de l'algorithme de segmentation sont décrites dans la figure 3.5 :

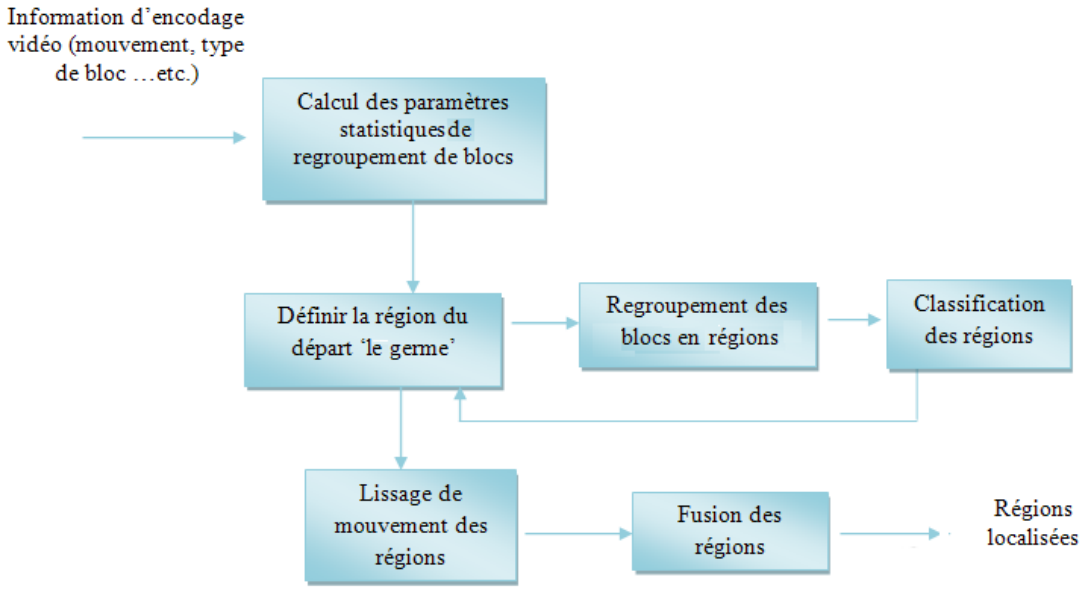


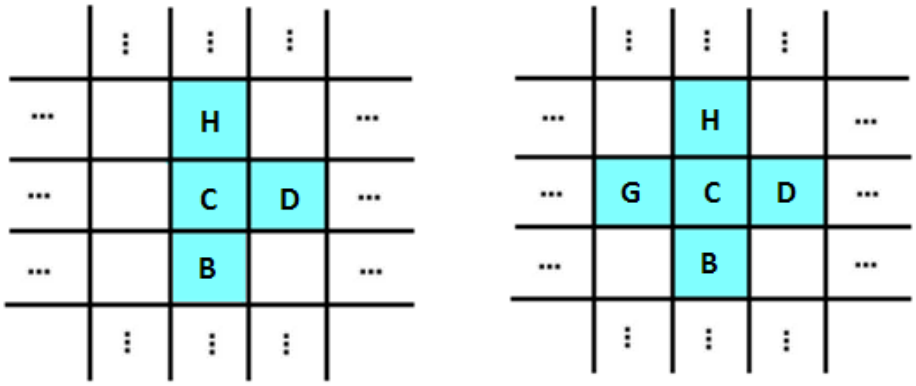
Figure 3.5 — Diagramme de l'algorithme de segmentation

3.3.2 Algorithme

- **Etape 1 : choix du germe initial**
  Le choix du germe est très important pour le reste de la segmentation, il représente un point de départ pour le regroupement ultérieur. Notons $\overrightarrow{MV}_i$, $i = 1, 2, \cdots, 5$ le vecteur de mouvement pour chaque bloc du germe, où 5 représente le nombre de blocs dans le germe. Le nombre de germes dépend du type de regroupement utilisé, il existe deux types de regroupement : un regroupement dans trois directions (germe avec 4 blocs) ou un regroupement dans quatre directions (germe avec 5 blocs).
  Dans le cadre de notre étude, le schéma basé sur les 4 directions a donné de meilleurs résultats que le regroupement sur 3 directions.
  Pour identifier le germe, nous parcourons tout les blocs de l'image (blocs candidats) et nous calculons la distance entre chaque bloc et ses quatres voisins. L'équation suivante est utilisée pour localiser le germe dans une image :

$$D_{germe} = \sum_{j \in germe} \|\overrightarrow{MV}_{bloc_{candidat}} - \overrightarrow{MV}_i\|) \tag{3.2}$$

$\overrightarrow{MV}_{bloc_{candidat}}$ représentre le mouvement du bloc candidat pour former le germe, voir la figure 3.6.



(a) regroupement à trois direction (b) regroupement à quatre direction

Figure 3.6 — Shémas de regroupement de blocs en régions

Dés lors qu'un bloc candidat est retenu (bloc ayant la distance minimale entre ses voisins comparé aux autres blocs candidats), les blocs voisins dans les quatre directions sont tout d'abord regroupés pour former le germe (les blocs en bleu clair sur la figure 3.7).

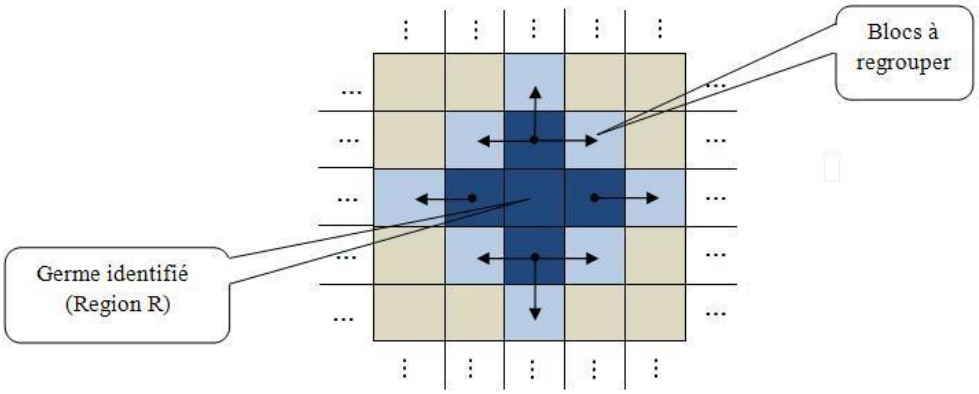


Figure 3.7 — Schéma de regroupement

- **Etape 2 : croissance de région**
  Soit R l'ensemble de vecteurs de mouvement de la région en croissance (au départ R = la région germe). Le critère de cohérence est basé sur la distance minimale entre les blocs de R ; il est adapté à chaque étape de regroupement.
  $\overrightarrow{MV}_{ext}$ est le vecteur de mouvement d'un bloc voisin de R, $\overrightarrow{MV}^{R}_{int}$ est le mouvement de chaque bloc à l'intérieur de R et $\overrightarrow{MV}^{R}_{Centre}$ la moyenne des vecteur du mouvement des blocs de la région. A chaque étape de regroupement $\overrightarrow{MV}^{R}_{Centre}$ est calculé pour déterminer le seuil de regroupement D_{TH} à cette étape (voir 3.3) :

$$D_{TH} = E\{\| \overrightarrow{MV}^R_{int} - \overrightarrow{MV}^R_{Centre} \|\} + D_{offset} \tag{3.3}$$

$$D_{offset} = min(\sigma^2_{MV}, T_{Max\_offset}) \tag{3.4}$$

σ^2_{MV} représente l'écart type du mouvement de blocs dans la région R. T_{Max_offset} un paramètre représente la variance maximale qu'on peut prendre en compte, obtenue par apprentissage sur la base de données (cf. chapitre 4, section §4.2.2)
D_{offset} représente la taille de la région R exprimé par la variance au sein de la région, utilisé lors du regroupement des blocs pour contrôler le regroupement rapide des blocs. Pour chaque bloc environnant, le regroupement est décidé lorsque la condition suivante est vérifiée :

$$D_i = \overrightarrow{MV}^R_{Centre} - \overrightarrow{MV}_{ext} \leqslant D_{TH} \tag{3.5}$$

- **Etape 3 : mise à jour du mouvement moyen de la région**
  Dès lors qu'un bloc est assigné à la région, la valeur $\overrightarrow{MV}_{Centre}$ est mise à jour en calculant à nouveau la moyenne des vecteurs de mouvement. Cette procédure est répétée pour tous les blocs voisins jusqu'à ce qu'il ne reste plus de blocs satisfaisant le critère de cohérence.
- **Etape 4 : recherche d'autres régions**
  Quand il n'y a plus de blocs à regrouper avec la région R. Un autre germe est identifié dans l'image et les étapes 1 à 3 sont répétées jusqu'à ce qu'il ne reste plus de germe dans l'image.
- **Etape 5 : fusion finale des régions**
  Les quatre étapes précédentes permettent de regrouper chaque germe identifié en régions. Cette étape permet d'affiner ce regroupement en fusionnant les régions similaires au sens d'un critère bien défini.
  Entre deux images successives, l'extraction des vecteurs de mouvement est assurée par le calcul de la différence absolue moyenne [8] lors de l'estimation de mouvement dans MPEG, ce qui entraine des vecteurs de mouvement bruités. Ces vecteurs génèrent des erreurs importantes lors de la segmentation. Pour les réduire, nous représentons chaque régions avec le vecteur de mouvement de son centre $\overrightarrow{MV}^R_{Centre}$.
  Pour toute paire de régions voisines, si la distance entre le mouvement des deux centres est inférieure à D_{min} donné par la formule (3.6), les deux régions seront fusionnées.

$$D_{min} = min(\sigma_{MV}, T_{Mov\_region}) \tag{3.6}$$

D_{min} est la distance minimale entre les blocs de la région en croissance, et T_{Mov_region} un parmètre obtenu par apprentissage sur la base de vidéos (cf. chapitre 4, section §4.2.2).

Résumé de l'algorithme

1. Localisation du germe en cherchant la distance minimale moyenne dans 4 directions pour chaque bloc de l'image, c'est le point de départ pour le regroupement.
2. En utilisant le seuil déterminé dans l'étape 2 (voir équation 3.3) et le germe bloc déterminé dans l'étape 1, nous procédons au regroupement des blocs voisins. Après chaque regroupement, les critères sont adaptés.

3. On répète les étape 1 à 3 jusqu'à ce qu'il ne reste plus de germe dans l'image.
4. On calcule le vecteur médian pour chaque région obtenue.
5. On calcule la distance entre chaque paire de régions voisines ; si la distance est inférieure au seuil déterminé par l'équation (3.6), les deux régions sont fusionnées.

La figure suivante donne un exemple de segmentation.

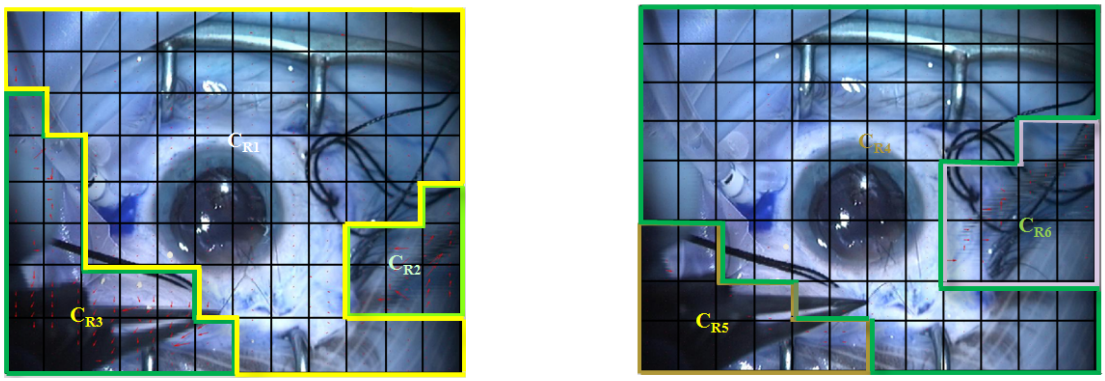


Figure 3.8 — Résultats obtenus en utilisant l'algorithme de croissance de régions à partir d'une région germe pour deux images de la séquence

Sur la figure 1.8, les blocs de vecteurs de mouvement similaires sont regroupés selon les critères d'homogénéité en 3 régions (R1, R2, R3).

Apres avoir segmenté les images constituant la séquence vidéo, nous nous intéressons ensuite à la trajectoire des régions au long de la vidéo. Pour cela nous avons mis en oeuvre un algorithme de suivi basé sur le filtre de Kalman [10], cela dans le but de prédire la position des régions qui se ressemblent au long de la séquence.

3.3.3 Algorithme de suivi (Filtre de Kalman)

L'approche développée dans la partie précédente permet d'extraire les régions en mouvement tout au long de la séquence en utilisant uniquement les images I. Pour déterminer la trajectoire de chaque région, nous proposons une technique de suivi qui nous permet d'effectuer des prédictions sur la future position de chaque région au sein de la prochaine image (la prochaine image I dans la séquence). La technique consiste à estimer la position du centre de la région dans l'image suivante, en utilisant un filtre de Kalman de deuxièmne ordre (Modèle à vitesse constante) en considérant que les actes sont à vitesse constante.

Le filtre de Kalman est un algorithme optimal et récursif pour l'estimation des paramètres [10]. Grâce à un modèle d'évolution des paramètres, cet algorithme calcule des prédictions et ajoute l'information, provenant de mesures, de façon optimale pour produire des estimations a posteriori des paramètres. Par rapport à la contrainte de temps-réel, des choix simples doivent être faits ; chaque région détectée est associée à un modèle d'évolution à vitesse constante, ceci conduit à un vecteur d'état X de 4 composantes :

$X = (x, y, v_x, v_y)$, avec (x, y) est le centre de la région. (v_x, v_y) est la vitesse du centre.

Le modèle d'évolution à vitesse constante conduit donc à la matrice d'évolution suivante :

$$A = A_t = \begin{pmatrix} 1 & 1 & T & 0 \\ 0 & 1 & 0 & T \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}$$

Les équations générales de prédiction et de filtrage de Kalman sont :

Equation de prédiction :

$\hat{X}_{t/t-1} = A_{t-1}\tilde{X}_{t-1/t-1} + B_{t-1}u_{t-1}$

$\hat{P}_{t/t-1} = A_{t-1}P_{t-1/t-1}A_{t-1}^T + Q$

Equation de correction (filtrage ou mise à jour) :

$\hat{X}_{t/t} = \hat{X}_{t/t-1} + G_t(s_t - G_t\hat{X}_{t/t-1})$

$G_t = P_{t/t-1}C_tT(R + C_tP_{t/t-1}C_tT)^2$

$P_{t/t} = (Id - G_tC_t)P_{t/t-1}$

Conditions initiales :

$\tilde{X}_{0/-1} = X_0$

$P_{0/-1} = P_0 = \diagup Id$

s_t : vecteur de mesures ; u_{t-1} : vecteur de commande ; $\hat{X}_{t/t-1}$: vecteur d'état prédit ; $\tilde{X}_{t/t}$: vecteur d'état estimé à posteriori ; X_0 : vecteur d'état initial ; A_{t-1} : matrice d'évolution ; B_{t-1} : matrice de commande ; C_t : matrice d'observation des mesures ; G_t : matrice de gain de kalman ; $P_{t/t-1}$: matrice de covariance prédite ; $P_{t/t}$: matrice de covariance estimée à posteriori ; P_0 : matrice de covariance initiale ; Q : matrice du bruit de modèle ; R : matrice de bruit d'observation ; Id : matrice d'identité ;

Voici les simplifications de notation qui découlent de notre modélisation ainsi que les hypothèses concernant certaines données.

Notre système de suivi de régions :

1. Il n'y a pas de vecteur de commande, soit $\forall t\ u_t = 0$
2. Nous observons à chaque instant t la position de la région, donc $s_t = [x_t, y_t]$ est le vecteur de mesure et C_t est la matrice d'observation.

$$C_t = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

3. Les mesures sont indépendantes et il n'y a que peu de bruit sur les mesures. Par conséquent, la matrice de bruit d'observation des mesures R est représentée par la matrice identité.
4. La matrice de bruit du modèle a été choisie diagonale et égale à la matrice identité.
5. Concernant les conditions initiales, le vecteur d'état initial X_0 est défini par les premières mesures ; la matrice de covariance initiale P_0 a été choisie diagonale et égale à la matrice identité ($P_0 = Id$).

   Ci-dessous les équations de prédiction et de filtrage de Kalman appliquées à notre modélisation, et simplifiées grâce aux hypothèses et aux choix précédemment décrits. Elles sont présentées dans l'ordre où elles sont calculées, pour l'instant t.

Equation de filtrage :

$\hat{X}_{t/t-1} = A\tilde{X}_{t-1/t-1}$

$\hat{P}_{t/t-1} = A_{t-1}P_{t-1/t-1}A_{t-1}^T + Q$

Equation de correction :

$G_t = P_{t/t-1}C_tT(R + C_tP_{t/t-1}C_tT)^-1$

$P_{t/t} = (Id - G_tC_t)P_{t/t-1}$

$\hat{X}_{t/t} = \hat{X}_{t/t-1} + G_t(s_t - C_t\hat{X}_{t/t-1})$

Les mesures étant disponibles uniquement lors du traitement de l'image courante, à l'instant t, nous commençons par estimer les variables en combinant les prédictions faites lors du traitement de l'image précédente à l'instant t-1, et les mesures provenant de l'image courante (obtenue par segmentation), à l'instant t (voir la figure 3.9).

En prenant la région R2 pour exemple, à l'instant t-1 nous avons le centre et la vitesse ; une prédiction par le filtre de Kalman nous amène à un $C_{R2prédit}$. Pour déterminer son correspondant dans l'image courante, nous calculons la distance minimale (distance euclidienne : d1, d2 et d3) par rapport au régions existantes (voir la figure 3.9). Ensuite, nous l'assignons à celle la plus proche, c'est ainsi que nous procédons pour suivre les régions.

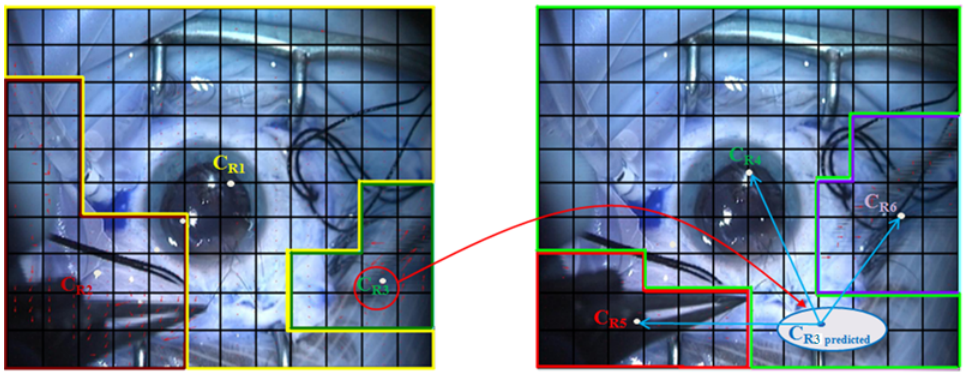


Figure 3.9 — Exemple de suivie de régions entre deux images de la séquence

Remarque : si la région disparait pendant le suivi, le filtre de Kalman associe la dernière apparition de la région à la région la plus proche dans l'image suivante. Le filtre de kalman est mis à jour pour suivre cette région.

Pour construire la signature, nous utilisons la trajectoire de chaque région. Cette trajectoire est caractérisée par le centre de chaque région (centre représenté par le germe (point de départ localisé lors de la segmentation et utilisé pour la prédiction de la région) (voir §3.3.1)), la vitesse du centre de la région ainsi que sa direction calculée par (§3.2).

Pour limiter les temps de calcul, seuls(les K régions les plus importantes en terme d'aire (typiquement $K = 5$) sont utilisées pour extraire la signature.

Notre vecteur descripteur a la forme suivante :

$Signature_{Vidéo} =< Centre_{i,k}, Vitesse_{i,k}, Direction_{ik} >$ Où i représente l'indice du l'image I et k l'indice de la région : $1 \leq k \leq K$.

Dans ce paragraphe, nous avons proposé une solution pour améliorer la représentation de la vidéo en ne considérant que les images I de la séquence. Nous retiendrons en particulier l'amélioration des temps de calcul (cf. chapitre 4). Toutefois, cette solution se traduit par

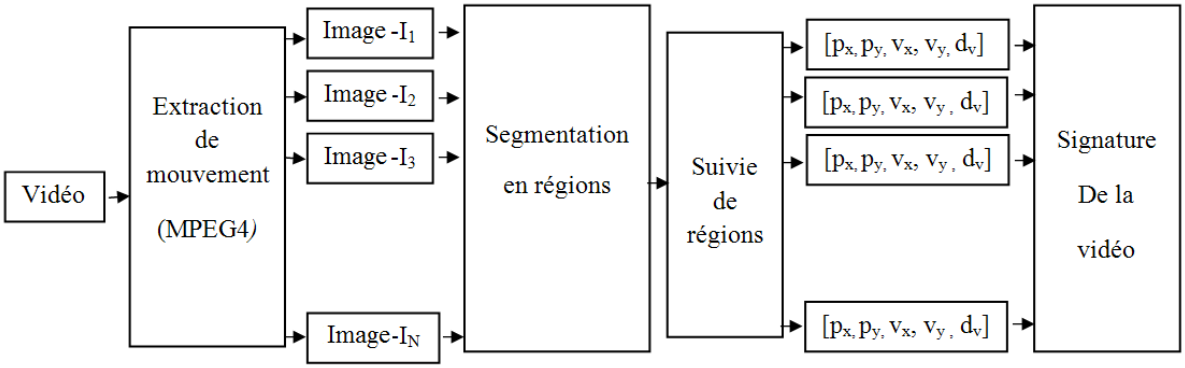


Figure 3.10 — Etapes de la caractérisation de la vidéo

une perte d'une partie de l'information située entre deux image I par rapport à la méthode du (§3.2) où nous utilisons toutes les images de la séquence. Les performances de retrouvaille sont moins bonnes comme nous le verrons au chapitre 4. Pour remédier à ce problème, nous proposons dans le paragraphe suivant un compromis entre le temps de calcul et une représentation plus complète de la vidéo.

3.4 Signatures basées sur le suivi des régions homogènes dans des GOPs sélectionnés

Afin de réduire la perte d'information engendrée par la méthode précédente, nous proposons dans cette approche, une solution basée sur le suivi des régions dans un résumé de la vidéo. Nous réalisons ce résumé en sélectionnant des GOPs (Group of pictures, cf. chapitre 2). Comme nous l'avons déja présenté dans la partie précédente, une image I est produite à chaque GOP (15 images) et elle constitue l'information principale du GOP. Pour construire le résumé de la vidéo, nous avons développé un algorithme de sélection de GOP, qui différencie les GOPs correspondant à des actions plus intéressantes, des GOPs sans action. Pour ce faire, nous avons développé tout d'abord un algorithme de comparaison entre images I. Il va nous permettre ensuite de faire la sélection des GOPs les plus pertinents et finalement d'obtenir le résumé de la vidéo qui nous servira pour construire la signature.

3.4.1 Mesure de similitude entre deux images I

Pour comparer deux images I, nous utilisons les histogrammes HSV.

Les images I sont extraites à intervalles réguliers : chaque GOP est limité par deux images I, nous avons donc une image I à chaque GOP (15 images). Un exemple des images I extraites dans un intervalle de 15 images d'une séquence vidéo est donné dans la figure 3.11 :

Nous comparons deux images I, I1 et I2, par la méthode suivante :

- Etape 1 : pour avoir plus de précision lors de la comparaison, I1 et I2 sont décomposées en macroblocs de taille $L \times L$ blocs.
- Etape 2 : pour chaque image I, m macroblocs sont sélectionnés aléatoirement, $1 \leqslant m \leqslant n$. n représente le nombre total de macroblocs de l'image I. Ce sont les mêmes macroblocs qui sont pris dans les deux images à comparer.
- Etape 3 : pour chaque macrobloc sélectionné, l'histogramme HSV des pixels est calculé.

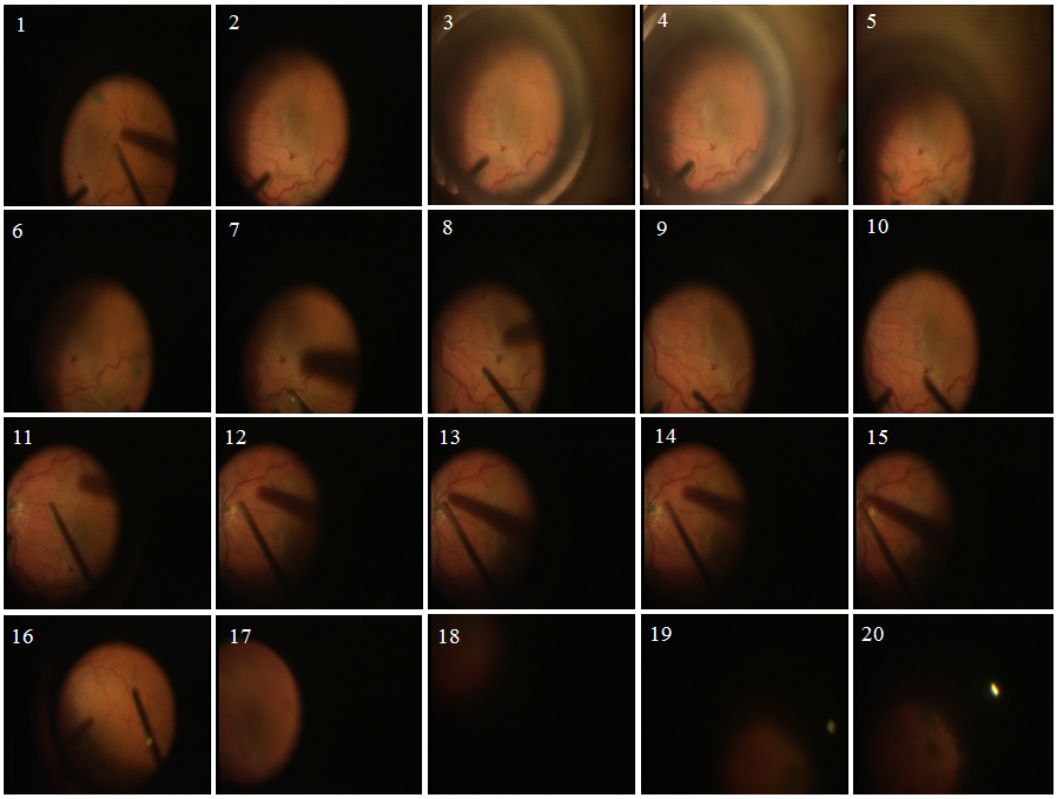


Figure 3.11 — Image I extraites dans un intervalle de 15 images d'une séquence vidéo

– Etape 4 : la distance entre deux macroblocs aux mêmes positions dans les images I1 et I2, est la distance entre leurs histogrammes HSV. La distance entre histogrammes HSV est calculée en utilisant la distance d'intersection d'histogramme [12] (formule 3.7) :

$$D(HSV_{macroblocI1}, HSV_{macroblocI2}) = \sum_{i=1}^{L} min[HSV_{macroblocI1}(i), HSV_{macroblocI2}] \quad (3.7)$$

Avec $HSV_{macroblocI1}$ qui représente l'histogramme HSV d'un macrobloc appartenant à I1.

– Etape 5 : deux macroblocs sont déclarés similaires si la distance mesurée est inférieure à un seuil $T_{Macrobloc}$, obtenu par apprentissage sur la base de vidéo (cf. chapitre 4, section §4.2.2).
– Etape 6 : deux images I1 et I2 sont considérées similaires si le nombre de macroblocs similaires est supérieur à la moité des macrobloc comparés.

La figure 3.12 présente l'application de l'algorithme sur les images I présentées dans la figure 3.11.

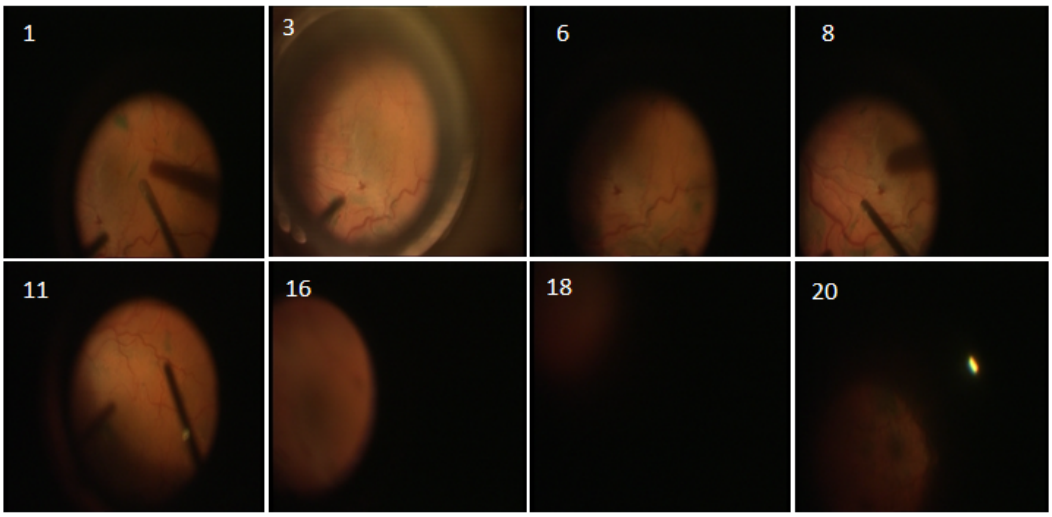


Figure 3.12 — Images I sélectionnées parmis les images présentées dans 3.11

3.4.2 Sélection de GOPs basée sur la similitude entre deux images I

L'étape de la sélection des GOPs est motivée par le besoin de générer un résumé de la vidéo efficace pour la construction de la signature. Cette étape permet d'avoir une description de la vidéo en ne gardant que les passages intéressants dans la séquence.

La construction du résumé commence par la sélection du premier GOP de la séquence suite à la sélection de la première image I. Pour le reste de la vidéo, la variation entre les images I consécutives est utilisée (§3.4.1). Les images I consécutives sont comparées, le résumé est constitué de la sélection de GOPs entre deux images I non similaires, consécutives dans le temps : elles correspondent a priori à un changement de plan, de scène, à une action, modification dans le contenu visuel de la vidéo. Le schéma de sélection de GOPs utilisé pour construire le résumé de la séquence est donnée dans la figure 3.13.

Comme nous le constatons sur la figure 3.12, le nombre d'images est réduit en utilisant uniquement les images. Ce nombre est encore réduit en utilisant la similarité et la sélection des images I où seuls les passages d'intérèt dans la séquence (nouvelles informations) sont gardés. Le but est de construire un résumé qui permette une caractérisation de la vidéo rapide et efficace.

Dans la figure 3.13, les images I1 et I2 sont similaires (pas de changement detecté), donc le GOP2 ne fera pas partie du résumé. Tandis que, si I1 et I3 ne sont pas similaires, le résumé est alors constitué des deux GOPs (GOP1 et GOP3).

Pour construire la signature de la vidéo, nous avons gardé la méthode développée au (§3.3). Afin de faciliter les calculs et considérer que les informations importantes, seules les K régions les plus grandes en terme d'aires sont utilisées pour extraire la signature.

Notre vecteur descripteur reste donc identique, mais en considérant les images des GOPs sélectionnés.

$Signature_{Résumé-Vidéo} =< Centre_{i,k}, Vitesse_{i,k}, Direction_{ik} >$ Où i représente l'indice des images dans les gops séléctionnés et k l'indice de la région : $1 \leq k \leq K$.

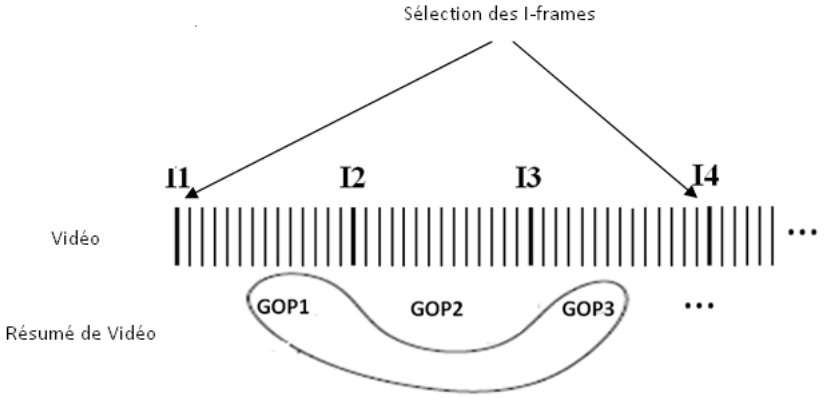


Figure 3.13 — Shéma de construction du résumé de la séquence

3.5 Caractérisation de l'information de résidu

Nos résultats (cf. chapitre 4) montrent que l'information de mouvement est pertinente pour caractériser nos vidéos. Mais d'autres informations sont présentes dans le flux MPEG4, nécessaires aussi pour décompresser les vidéos, et donc tout aussi porteuses d'informations : il s'agit des informations liées aux valeurs des points dans l'image (niveaux de couleur/niveaux de gris). Cette information se retrouve en partie dans les résidus qui sont calculés dans la norme. Le résidu du bloc courant, appelé aussi erreur de prédiction, est la différence entre la valeur du bloc prédit et celle du bloc courant (cf. chapitre 2).

Nous allons utiliser cette information pour enrichir nos signatures basées sur l'analyse du mouvement. L'idée est de modéliser la distribution des coefficients de résidus par une loi statistique, dont nous pourrons utiliser les paramètres comme paramètres supplémentaires dans nos signatures. De nombreux modèles statistiques ont été proposés pour modéliser cette information de résidu. Pao et al proposent d'utiliser un modèle Laplacien [13], mais Wang et al ont démontré que les dernières avancées dans le domaine de l'estimation de mouvement (MPEG4) produisaient un résidu proche d'un bruit gaussien aléatoire. L'utilisation d'une modélisation par une gaussienne généralisée semble donc être la plus appropriée.

3.5.1 Loi gaussienne généralisée

La loi gaussienne généralisée est une loi centrée qui dérive de la loi normale. La loi normale centrée, n'étant définie que par un paramètre, ne suffit pas à représenter précisément la distribution. Par contre la loi gaussienne généralisée est paramétrée par deux valeurs :

- α : le paramètre d'échelle, qui correspond à l'écart-type dans le cas d'une loi gaussienne classique.
- β : le paramètre de forme, qui est inversement proportionnel au taux de décroissance du pic (il vaut 2 dans le cas d'une gaussienne)

L'expression de sa densité est la suivante (équation 3.8) :

$$p(x;\alpha,\beta) = \frac{\beta}{2\alpha\Gamma(\frac{1}{\beta})} e^{-\left(\frac{|x|}{\alpha}\right)^{\beta}} \tag{3.8}$$

où $\Gamma(z) = \int_0^\infty e^{-t}t^{z-1}dt$, $z > 0$ est la fonction gamma. Une méthode pour obtenir une approximation de la fonction Γ avec une précision de 2×10^{-10} $\forall z > 0$ est donnée dans

Numerical Recipes in C [14]. Des exemples de densités de lois gaussiennes généralisées sont données figure 3.14.

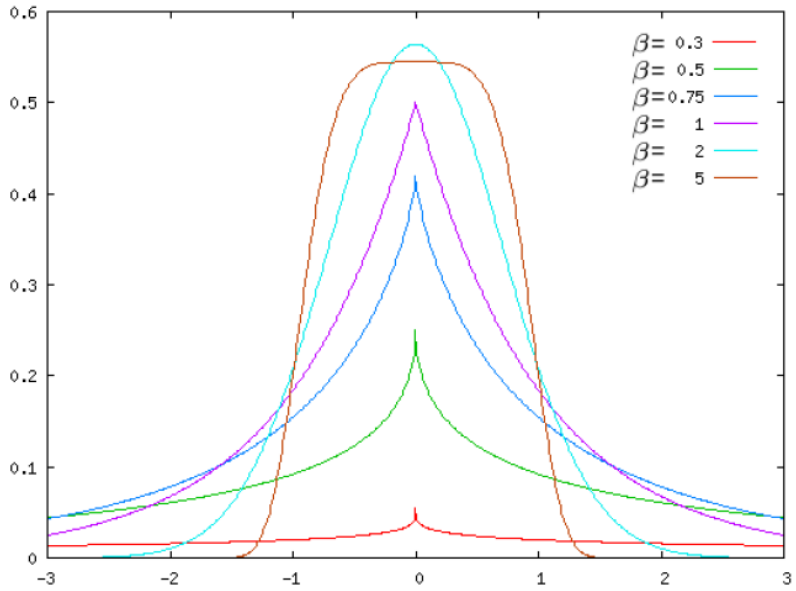


Figure 3.14 — Exemples de densités de lois gaussiennes généralisées (différentes valeurs pour β / $\alpha = 1$)

3.5.2 Estimation des paramètres par maximum de vraisemblance

Nous cherchons à calculer un estimateur du maximum de vraisemblance pour cette distribution : soit $\mathbf{x}= (x_1, ..., x_L)$ le vecteur des coefficients de résidu pour une image donnée où L est le nombre de macrobloc. Nous supposons les données x_i indépendantes. Alors Varanasi et Aazhang [15] ont montré que l'estimateur du maximum de vraisemblance $(\hat{\alpha}, \hat{\beta})$ est donné par le système d'équation suivant :

$$\hat{\alpha} = \left(\frac{\hat{\beta}}{L} \sum_{i=1}^{L} |x_i|^{\hat{\beta}} \right)^{\frac{1}{\hat{\beta}}} \tag{3.9}$$

$$1 + \frac{F(\frac{1}{\hat{\beta}})}{\hat{\beta}} - \frac{\sum_{i=1}^{L} |x_i|^{\hat{\beta}} \log |x_i|}{\sum_{i=1}^{L} |x_i|^{\hat{\beta}}} + \frac{\log \left(\frac{\hat{\beta}}{L} \sum_{i=1}^{L} |x_i|^{\hat{\beta}} \right)}{\hat{\beta}} = 0 \tag{3.10}$$

où la fonction digamma F est définie par $F(z) = \frac{\Gamma'(z)}{\Gamma(z)}$. La valeur de $\hat{\beta}$ doit donc d'abord être retrouvée par un algorithme de recherche de racines. L'algorithme itératif de Newton-Raphson [16] converge efficacement vers l'unique solution, à condition d'être bien initialisé. Pour déterminer une solution initiale, Do et Vetterli [17] proposent une méthode qui permet de faire converger Newton-Raphson en typiquement 3 itérations avec une précision de 10^{-6}. Cette méthode consiste à faire correspondre l'écart type et la moyenne, nommés moments de résidu de l'image (m_1 et m_2), avec ceux de la gaussienne généralisée

recherchée. Do a montré que le rapport entre la moyenne et l'écart-type d'une gaussienne généralisée est une fonction monotone F_M de β, définie par l'équation 3.11 [17] :

$$F_M(\beta) = \frac{\Gamma(\frac{2}{\beta})}{\sqrt{\Gamma(\frac{1}{\beta})\Gamma(\frac{3}{\beta})}} \tag{3.11}$$

Une estimation initiale $\bar{\beta}$ pour β est donc donnée par l'équation 3.12 :

$$\bar{\beta} = F_M^{-1}\left(\frac{m_1}{\sqrt{m_2}}\right) \tag{3.12}$$

L'algorithme de Newton-Raphson peut être utilisé pour la recherche de la racine de la fonction monotone suivante :

$$x \mapsto F_M(x) - \frac{m_1}{\sqrt{m_2}} \tag{3.13}$$

3.5.3 Algorithme de Newton-Raphson et algorithme de Newton-Raphson robuste

Pour calculer les racines d'une fonction f par la méthode de Newton-Raphson, il faut connaître une expression de la dérivée f' de f. Le principe de l'algorithme est le suivant :

(i) une solution initiale x_0 est fournie

(ii) tant que $\frac{f(x_k)}{f'(x_k)} > \varepsilon : x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$

La dérivée du premier membre de l'équation 3.10, pour le calcul de $\hat{\beta}$ est donnée par [17]. Elle fait intervenir la fonction trigamma $\digamma'$, dérivée de la fonction $\digamma$. Des approximations efficaces de ces deux fonctions sont également disponibles.

La dérivée de l'équation 3.13, pour le calcul de $\bar{\beta}$ est déterminée de la manière suivante :

– Soit $lF_M(\beta) = \log F_M(\beta)$; $lF_M(\beta)$ est défini par l'équation 3.14.

$$lF_M(\beta) = \log(\Gamma(\frac{2}{\beta})) - \frac{\log(\Gamma(\frac{1}{\beta})) + \log(\Gamma(\frac{3}{\beta}))}{2} \tag{3.14}$$

– Soit dlF_M la dérivée de lF_M par rapport a β. dlF_M est déterminée par l'équation 3.15.

$$dlF_M(\beta) = \frac{\frac{1}{2} \times \digamma(\frac{1}{\beta}) + \frac{3}{2} \times \digamma(\frac{3}{\beta}) - 2 \times \digamma(\frac{2}{\beta})}{\beta^2} \tag{3.15}$$

– Finalement, la dérivée de $F_M(\beta)$, $dF_M(\beta)$, s'obtient par l'équation 3.16.

$$dF_M(\beta) = F_M(\beta) \times dlF_M(\beta) \tag{3.16}$$

Dans le cas d'images médicales, nous constatons que pour certaines images l'algorithme de Newton-Raphson peut diverger. Cela se produit lorsque le rapport $\frac{f(x_k)}{f'(x_k)}$ (point (ii) de l'algorithme) est trop important. Il se peut alors qu'après mise à jour, x_{k+1} soit négatif ou proche de 0, après quoi l'algorithme diverge. Ceci nous a conduit à proposer une modification de l'algorithme initial de Newton-Raphson pour le rendre robuste [18].

Nous couplons l'algorithme de Newton-Raphson à un algorithme de bissection (qui converge moins rapidement, mais est assuré de converger). Le principe de la méthode de bissection est le suivant :

1. on recherche dans un premier temps a et $b > a$ tels que $f(a)f(b) < 0$; cette première étape consiste donc à encadrer la solution cherchée,
2. on choisit un point c entre a et b, typiquement $c = \frac{a+b}{2}$, puis on calcule $f(c)$,
3. si $f(a)f(c) < 0$ la solution est entre a et c, sinon elle est entre c et b,
4. l'intervalle $[a, b]$ est donc remplacé par $[a, c]$ ou $[c, b]$ suivant le cas et le processus est itéré jusqu'à convergence.

A chaque itération, nous décidons s'il faut appliquer une étape de l'algorithme de bissection ou une étape de l'algorithme de Newton-Raphson, ces deux méthodes faisant bien sûr évoluer le même point. Le choix entre ces deux étapes se fait en fonction de $|\frac{f(x_k)}{f'(x_k)}|$. Si $|\frac{f(x_k)}{f'(x_k)}| < \epsilon$, $\epsilon > 0$ et $\epsilon << (b - a)$, nous effectuons une étape de l'algorithme de bissection, sinon nous effectuons une étape de l'algorithme de Newton-Raphson (nous avons choisi $\epsilon = 10^{-6}$). De même, si $x_k - \frac{f(x_k)}{f'(x_k)}$ est en dehors de l'intervalle $[a; b]$, alors x_{k+1} est déterminé par une étape de l'algorithme de bissection.

Il faut définir un nouveau critère d'arrêt pour l'algorithme hybride, un critère qui soit calculable pour les deux méthodes. Le critère suivant a été utilisé : $x_{courant} - x_{precedent} < seuil$. Pour la méthode de bissection cette différence vaut $c = \frac{b-a}{2}$, pour la méthode de Newton-Raphson il vaut $\frac{f(x)}{f'(x)}$. L'intervalle initial pour la bissection est $[\epsilon; C]$. Nous avons choisi $C = 5$, qui n'est jamais atteint. En effet, la valeur de β est généralement comprise entre 0 et 2 (la forme des distributions est proche d'un Laplacien).

3.5.4 Adaptation aux résidus des vidéos compressées étudiées

Une approximation par gaussienne généralisée des résidus extraits des vidéos que nous utilisons, semble convenir. Les paramètres de la gaussienne généralisée sont recherchés en utilisant la méthode présentée ci-dessus (voir §3.5.2). Pour montrer l'adéquation de la gaussienne généralisée à la distribution des coefficients de résidus, nous avons superposé les histogrammes des coefficients des résidus avec les densités des gaussiennes généralisées d'une vidéo de chirurgie de la rétine.

Dans la figure 3.15, le peu de données de résidu par image (une valeur par bloc) nous impose d'avoir un faible nombre de classes.

3.6 Combinaison des signatures

Afin de produire des signatures efficaces et plus riches en terme d'information que celles obtenues à partir de l'information de mouvement, nous avons donc combiné chacune des signatures exposées précédemment (§3.2, §3.3, §3.4) avec l'information de résidu caractérisée par les deux paramètres α et β. La figure 3.16 illuste cette combination :

Nos vecteurs descripteurs suivant les 3 méthodes peuvent être écrits de la manière suivante :

Méthode 1 (§3.2) :

$$Signature_{Vidéo} =< Direction_i, Angle_i, Intensité_i, \alpha_i, \beta_i > . \tag{3.17}$$

Où $1 \leq i \leq N$ représente le nombre d'images de la séquence.

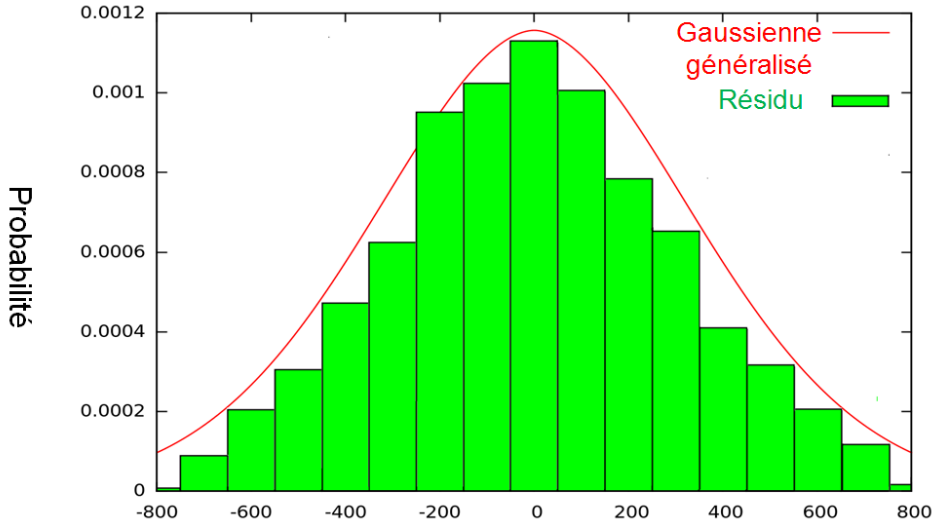


Figure 3.15 — Modélisation de l'histogramme des coefficients de résidu par une gaussienne généralisée (ses parametre $\alpha = 486.385$ et $\beta = 1.95$)

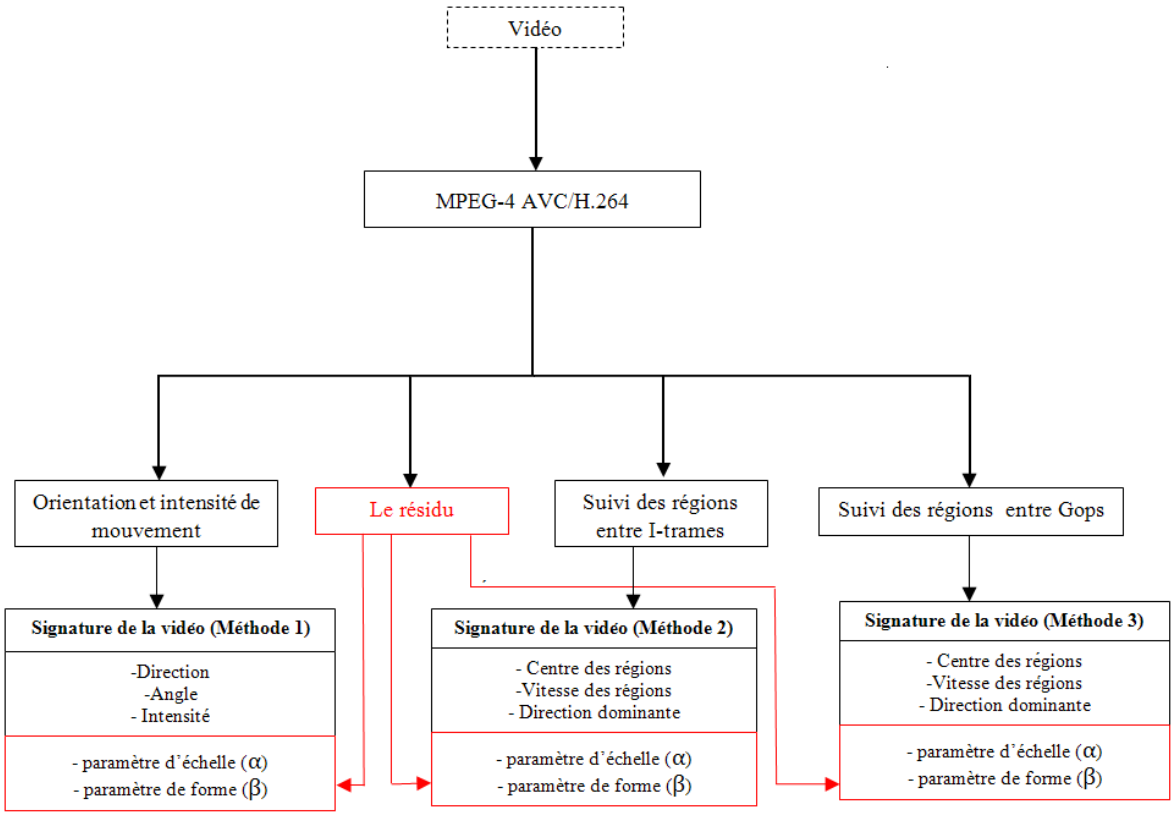


Figure 3.16 — Etape de construction des signatures

Méthode 2 (§3.3) :

$$Signature_{Vidéo} =< Centre_{i,k}, Vitesse_{i,k}, Direction_{ik}, \alpha_{ik}, \beta_{ik} > \quad (3.18)$$

Où i représente l'indice de l'image I et k l'indice de la région : $1 \leq k \leq K$.

Méthode 3 (§3.4) :

$$Signature_{Résumé-Vidéo} =< Centre_{i,k}, Vitesse_{i,k}, Direction_{ik}, \alpha_{ik}, \beta_{ik} > \quad (3.19)$$

Où i représente l'indice de l'image I dans les GOPs sélectionnés et k l'indice de la région : $1 \leq k \leq K$.

3.7 Mesures de distance entre deux signatures : définition générale

Nous devons comparer des séquences vidéos chirurgicales, qui n'ont pas forcément les mêmes durées pour des chirurgies identiques, que ce soit globalement, ou dans les différentes phases des chirurgies. Notons que cette remarque peut aussi s'appliquer à des séquences d'examens vidéoendoscopiques par exemple.

La nature spatio-temporelle des vidéos nécessite donc de spécifier une mesure qui tienne compte des propriétés à la fois spatiales et temporelles de la séquence d'images. Une méthode communément utilisée pour comparer deux séquences numériques est de chercher l'ensemble des déformations de coût minimal nécessaires à l'alignement d'une séquence sur l'autre. Les déformations locales autorisées (ou opérations d'édition) sont l'insertion ou la suppression d'un élément dans une séquence et le remplacement d'un élément d'une séquence par un autre. Celles-ci définissent un modèle de déformations locales et nous verrons plus loin comment ces principes sont mis en oeuvre pour notre cadre applicatif spécifique.

Trois types de signatures sont proposées. Dans la première nous cherchons à caractériser la vidéo dans sa globalité : chaque vidéo est représentée par un vecteur de 5 dimensions (direction, angle et intensité, paramètre d'échelle (α), paramètre de forme (β)). Pour cette signature, la mesure de similitude entre deux séquences est calculée en utilisant la distance FDTW (Fast dynamique Time Warping) que nous présentons dans la section §3.7.2. Les deux autre méthodes étant basées sur la trajectoire des régions dominantes, chaque vidéo est représentée par un vecteur de signature de 7 dimensions (position des régions, vitesses, angles des vitesses, paramètres d'échelle (α), paramètre de forme (β))). La distance entre deux vidéos est alors calculée en utilisant une nouvelle méthode, appelée EFDTW (Extended Fast Dynamic Time Warping) qui représente une extension de la FDTW dans le domaine multidimentionnel, où chaque dimension est représentée par la trajectoire de l'une des régions localisées (§3.8.2).

Nous présentons dans un premier temps l'algorithme classique, appelé alignement dynamique temporel, ou Dynamic Time Warping (DTW), qui permet d'obtenir efficacement l'ensemble de déformations de coût minimal. Cet algorithme est à l'origine de l'algorithme rapide FDTW que nous utilisons. En (§3.7.3) nous présentons la distance EMD (Earth Mover's Distance) qui nous a conduit à la distance EFDTW, en la combinant avec l'algorithme FDTW.

3.7.1 DTW (Dynamic Time Warping)

Sakoe et Chiba [19] posent le problème du calcul de l'ensemble de déformations de coût minimal entre deux séquences en terme de mesure de la dissimilarité entre les séquences considérées.

Soient Q et C deux séquences de tailles respectives (m, l) dont on souhaite connaître la similarité :

$$Q = q_1, q_2, ..., q_m, C = c_1, c_2, ..., c_l \tag{3.20}$$

On commence par construire une matrice S de correspondance entre les deux séquences, de taille $m \times l$ que l'on définit comme :

$$S_{i,j} = d(q_i, c_j),\ \forall (i,j) \in [1;m] \times [1;l] \tag{3.21}$$

Où d représente la distance euclidienne.

On définit un chemin w, de taille K, comme une suite de paires $w(i,j) \in S_{i,j}\ avec\ (i,j) \in [1;m] \times [1;l]$, éléments de S vérifiant les contraintes aux bords :

$$w(1) = S(1,1),\ w(K) = S(m,l). \tag{3.22}$$

Le chemin W considéré doit vérifier les propriétés suivantes :

$$Continuité : Si\ w_k(i,j)\ et\ (w_{k-1}(i',j')),\ alors\ (|i-i'| \leq 1)\ et\ (|j-j'| \leq 1)) \tag{3.23}$$

$$Monotonie : Si\ w_k(i,j)\ et\ (w_{k-1}(i',j'))\ alors\ (|i-i'| \geq 0)\ et\ (|j-j'| \geq 0))\ et\ |i-i'| + |j-j'| \neq 0 \tag{3.24}$$

On appelle transition un ensemble de deux paires d'indices consécutifs au sein d'un chemin. On peut noter que la propriétés de continuité ci-dessus énoncée limite les chemins acceptables à n'utiliser que des transitions horizontales (notées $(1,0)$), verticales (notées $(0,1)$) et diagonales (notées $(1,1)$).

Il en découle la formule de récurrence suivante :

$$\forall (i,j) \in [1;m] \times [1;l], \Gamma_{i,j} = min \begin{cases} S(i-1,j) \\ S(i-1,j-1) \\ S(i,j+1) \end{cases} \tag{3.25}$$

où $\Gamma_{i,j}$ représente la distance cumulée calculée en utilisant la matrice S constituée de ses i premières lignes et j premières colonnes ou, en d'autres termes, le coût de l'alignement des séquences $q_1, ..., q_i$ et $c_1, ..., c_j$.

Pour calculer la distance DTW entre les deux séquences, le chemin w est calculé de la manière suivante :

1. $w(1) = \Gamma(1,1)$ et $w(k) = \Gamma(m,l)$.
2. Pour chaque élément $w(i,j)$ de la matrice Γ, le choix des prédécesseurs est limité à $(w(i-1,j), w(i,j-1), w(i-1,j-1))$.
3. Le chemin est déterminé en allant de $w(m,l)$ vers $w(1,1)$.
4. La distance minimale entre les deux séquences est donnée par la somme des éléments du chemin w.

Nous donnons ci-dessous la distance DTW pour les deux séquences V_A et V_B, caractérisées par les valeurs dans 3.17, où chacune des images de la séquence est représentée par une valeur. Le chemin est donné dans la figure 3.18 :

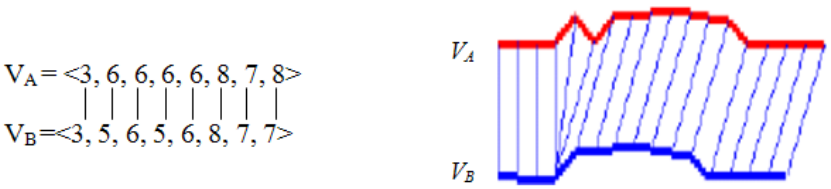


Figure 3.17 — Schéma illustrant l'alignement des deux séquences

La distance DTW est de 0+1+1+2+2+2+2+3 = 13.

V_B/V_A	3	6	8	7	8
3	**0**	3	8	12	17
5	2	**1**	4	6	9
6	5	**1**	3	4	6
5	7	**2**	4	5	7
6	10	**2**	4	5	7
8	15	4	**2**	3	3
7	19	5	3	**2**	**3**

Figure 3.18 — Tableau de distances cumulées

Pour éviter de calculer plusieurs fois chaque terme d'indice (i, j), et avoir la distance en même temps que le calcul de la matrice cumulée Γ, Sakoe et Chiba [19] proposent d'utiliser la programmation dynamique et ce que nous avons utilisé pour calculer la distance DTW.

L'algorithme consiste à calculer d'une façon récursive la distance cumulée minimum pour chaque point (i,j) en tenant compte des contraintes citées ci-dessus.

---

algorithme de la distance DTW Algorithme de calcul de la distance DTW. Il détermine le chemin optimal et son coût, pour la matrice S de similarité entre les séquences considérées.

Entrées : S[1..m][1..l]

float $\Gamma[0..m][0..l]$

int i, j

$\Gamma[0][0] \leftarrow 0$

$\Gamma[0][1..l] \leftarrow +\infty$

$\Gamma[1..m][0] \leftarrow +\infty$

pour i = 1 à m faire

pour j = 1 à l faire

$\Gamma[i][j] \leftarrow S[i][j] + min(\Gamma[i-1][j], \Gamma[i][j-1], \Gamma[i-1][j-1])$

fin pour

fin pour

renvoi $\Gamma[m][l]$

Cet algorithme a une complexité temporelle en $\mathcal{O}(m \times l)$ et une complexité spatiale en $\mathcal{O}$ $(m \times l)$. Cette dernière peut toutefois aisément être réduite à $\mathcal{O}(l)$ car il suffit, lorsque l'on traite la ligne d'indice i, de stocker les lignes d'indices i-1 et i de la matrice Γ, les précédentes n'étant plus utiles. De la même manière, si l'on a $m \ll l$, il sera judicieux de remplir la matrice Γ colonne par colonne et limiter la complexité spatiale à $\mathcal{O}(m)$.

Nous avons utilisé cette méthode pour mesurer la distance entre deux séquences. Une telle procédure de recherche directe serait très lente (voir résultats, cf. chapitre 4) à cause des nombreuses possibilités de recherche de chemin optimal, surtout lorsqu'on travaille avec de longues séquences. Pour pallier ce problème, nous avons utilisé une version modifiée de la DTW appelée Fast DTW (FDTW) pour l'alignement dynamique.

3.7.2 FDTW (Fast Dynamic Times Warping)

Le principe est ici de diminuer la complexité du calcul de la DTW par une méthode de recherche de chemin optimal entre les séquences plus efficace. Dans cet algorithme, on définit une enveloppe qui sert à limiter la recherche. Sakoe et Chiba [19] utilisent une enveloppe sous forme de bande. Cette idée a ensuite été améliorée par Keogh et Ratanamahatana [20] en appliquant une contrainte globale au calcul de la DTW : ils imposent à la bande de rester étroite, de largeur 2r + 1 autour de la diagonale (voir la figure 3.19). Le paramètre r dépend de la longeur de C, $r = 0.1|C|$, illustré dans [21].

Cette contrainte équivaut à ne chercher le chemin dans la matrice Γ que pour les termes d'indices (i, j) tels que :

$$\{q_j, |i - j| \in [0, R]\} \tag{3.26}$$

Afin de calculer la distance FDTW, les minima et maxima (enveloppe formée par U et L) de la séquence requête Q sont construits (voir formule 3.27 et 3.28). C'est l'ensemble des éléments de Q susceptibles d'être mis en correspondance avec c_i.

$$U : \forall i \in [1; m]; u_i = max\{Q_{i-r}, Q_{i+r}\} \tag{3.27}$$

$$L : \forall i \in [1; m]; l_i = min\{Q_{i-r}, Q_{i+r}\} \tag{3.28}$$

$i - r, i + r$ représentent la bande.

On voit notamment ici l'intérêt de se restreindre à une bande autour de la diagonale pour $D_{LB_{Keogh}}$ qui offre ainsi une mesure précise et plus rapide. Cette figure est inspirée de [20].

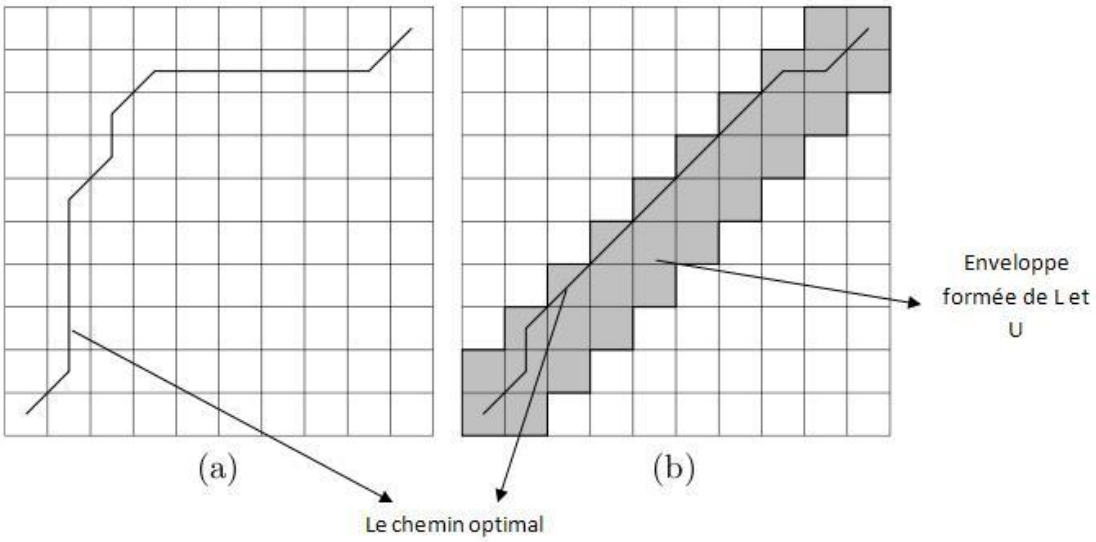


Figure 3.19 — Alignement dynamique et chemins. Deux exemples de chemins pour l'alignement dynamique : le premier (a) correspond aux contraintes classiques telles qu'énoncées par l'équation 1.28, le deuxième (b) correspond à un chemin contraint globalement par une bande de Sakoe-Chiba de largeur r = 1.

Pour mieux comprendre l'utilité de cette bande, nous expliquons le principe dans la figure 3.20 :

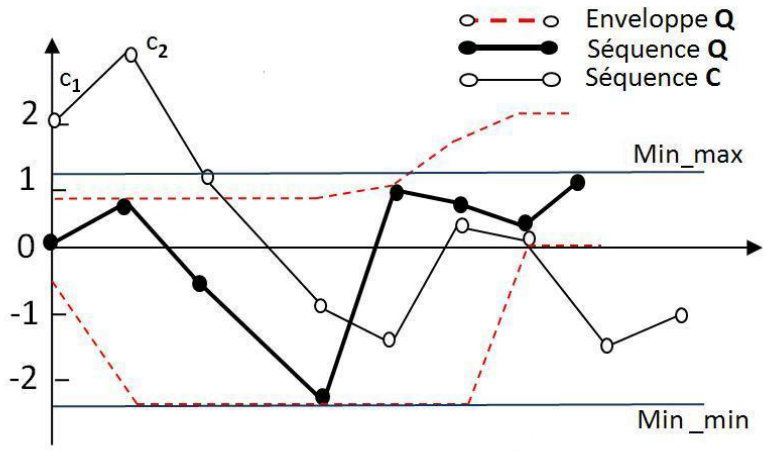


Figure 3.20 — Principe intuitif du calcul de la FTDW

Le principe de la FTDW est basé sur l'observation suivante : après le calcul de la bande par les deux formules données ci-dessus, tout élément au-dessus du minimum des deux maxima des deux séquences devrait contribuer au calcul de la distance. Par exemple, les deux élément c_1 et c_2 sur la figure sont supérieurs à la valeur minimale des deux maximas (Min_{max}), donc ils doivent contribuer au calcul de la distance. En fonction de la propriété de continuité du trajet de déformation, ces éléments doivent être mis en correspondance avec au moins un élément de Q et bien évidemment avec celui qui a la distance minimale. Ceci nous évite d'aligner les points de C avec d'autres points (éviter le balayage de toute la matrice de distance comme

c'est le cas pour la DTW (voir les deux manières de recherche du chemin optimal dans la figure 3.19). La même observation pour les éléments au-dessus du minimum des deux minimas (Min_{min} dans la figure 3.20)

Pour pouvoir comparer les séquences C, U et L, celles-ci doivent être de même taille. Pour rendre cela possible, les auteurs proposent de ré-échantillonner la séquence C. Ainsi, dans la suite de cette partie, la séquence C sera supposée de longueur m. Il faut toutefois noter que les bornes inférieures proposées par la suite minorent la DTW calculée, non pas entre les séquences Q et C, mais entre la séquence Q et la version de la séquence C ré-échantillonnée. La distance $D_{LB_{Keogh}}$ est calculée en approchant l'une des deux séquences, ici Q, par son enveloppe. Le principe est d'estimer la similarité entre les deux séquences par la somme des distances au carré entre chacun des éléments de C et le plus proche point dans l'intervalle $[l_i\,;\,u_i]$ (voir la figure 3.19).

On définit alors :

$$D_{LB_{Keogh}} = \sum_{i=1}^{n} \begin{cases} (c_i - u_i)^2 & si \quad c_i > u_i \\ (c_i - l_i)^2 & si \quad c_i < l_i \\ 0 & sinon \end{cases}$$

Keogh et Ratanamahatana montrent dans [20] que $D_{LB_{Keogh}}(Q,C) \leq DTW(Q,C)$.

Sur la figure 3.21, la requête est dessinée en trait fin et la séquence de la base à laquelle elle est comparée en trait gras. l'aire colorée en gris correspond à la mesure relative à la borne inférieure considérée. La contrainte globale considérée est une bande de Sakoe-Chiba [19], menant à une enveloppe de largeur fixe 2r + 1 telle qu'indiquée :

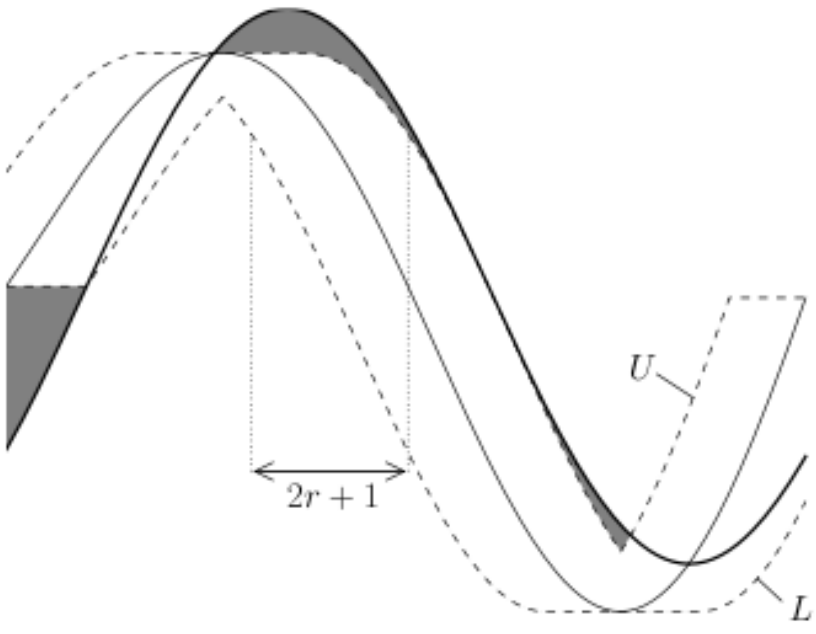


Figure 3.21 — Enveloppe pour la DTW : la borne supérieure et inférieure pour la séquence requête

3.7.3 Earth Mover's Distance (EMD)

Nous présentons dans ce paragraphe, d'une manière générale, la distance EMD entre deux séquences d'images Q et C. Nous verrons l'utilité de l'EMD dans la section (§3.8.2).

L'EMD (Earth Mover's Distance) est la quantité minimale de travail nécessaire pour changer une séquence en une autre séquence. Soient deux séquences à comparer Q et C, représentées par les vecteurs $Q = q_i$ et $C = c_j$, $1 \leq i \leq m$ et $1 \leq j \leq l$. Selon [22], la première étape consiste à trouver l'ensemble des portions de contenu $f_{i,j}$ de la distribution à transporter de la composante i vers la composante j qui minimise le coût (travail) suivant :

$$EMD(Q,C) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{l} f_{i,j}D}{\sum_{i=1}^{m}\sum_{j=1}^{l} f_{i,j}} \tag{3.29}$$

Où D représente la matrice de distance entre Q et C et $f_{i,j} \geq 0$ les déplacements entre Q et C minimisant les equations suivantes :

$$\sum_{i=1}^{m} f_{i,j} \leq W_{Q_i}, \sum_{j=1}^{l} f_{i,j} \leq W_{C_j} \tag{3.30}$$

$$\sum_{i=1}^{m}\sum_{j=1}^{l} f_{i,j} = min(\sum_{i=1}^{l} W_{Q_i}, \sum_{j=1}^{m} W_{C_j}) \tag{3.31}$$

Où W_{Q_i} et W_{C_j} représentent les points associés à chaque composants des deux séquences. Puisque nous avons déja optimiser les distances avec les algorithmes génétiques (voir §3.9, équation (3.46)), donc nous avons fixé les points à la valeur 1 : $W_{Q_i} = 1$ et $W_{C_j} = 1$.

3.8 Mesures de distance entre deux signatures : adaptation aux signatures présentées

3.8.1 Mesures de distance entre signatures basées sur l'intensité et l'orientation de mouvement

Pour comparer deux vidéos, nous comparons leurs signatures présentées dans (§3.6). Pour cela, nous devons établir une mesure de distance entre elles. Nous avons utilisé la DTW (Dynamic time warping) dans un premier temps. Considérons deux vidéos Q et C, $Q = \{q_i\}$ et $C = \{c_j\}$ $1 \leq i \leq m$ et $1 \leq j \leq l$. Chaque séquence est représentée par son vecteur caractéristique $V_Q = \{V_{Q_i}\}, 1 \leq i \leq m$, and $V_C = \{V_{C_j}\}, 1 \leq j \leq l$ avec :

$$V_{Q_i} = \{V_{Qi,v}\} \quad 1 \leq v \leq 5 \tag{3.32}$$

$$V_{C_j} = \{V_{Cj,v}\} \quad 1 \leq v \leq 5 \tag{3.33}$$

Où $1 \leq v \leq 5$, représente les cinq composantes de la signature (direction, angle, intensité et α, β) de l'image Q_i de la séquence Q (respectivement de l'image C_j de la séquence C) (voir §3.6).

La distance entre les deux séquences d'images se traduit par la distance entre leurs vecteurs caractéristiques et se traduit par la formule suivante :

$$D(V_Q, V_C) = \sum_{v=1}^{5} \gamma_v DTW(V_{Qi,v}, V_{Cj,v}) \tag{3.34}$$

Les γ_v sont des poids d'ajustement utilisés pour favoriser certaines composants de la signature par rapport aux autres (voir section §3.9)

DTW représente la distance qui se calcule suivant les étapes déja présentées dans la section (§3.7.1) Un exemple de calcul est donné par la figure 3.18.

Afin de diminuer la complexité de la DTW, et passer à une méthode de recherche de chemin optimal moins grossière entre les séquences, nous avons utilisé la FDTW (voir §3.7.2).

3.8.2 Mesure de distance entre signatures basées sur le suivi des régions

Dans cette partie, nous procédons de la même manière que dans (§3.8.1) pour calculer la distance entre les trajections de paires de régions. Quant à la distance entre deux vidéos (avec le nombre de régions considérées, typiquement $K = 5$), nous utilisons EFDTW qui est la combinaison de la FDTW (Fast Dybamic Time Warping) et l'EMD (Earth mover's distance).

Nous considérons deux vidéos $Q = \{q_i\}$ et $C = \{c_j\}$ $1 \leq i \leq m$ et $1 \leq j \leq l$. Chaque séquence est représentée par son vecteur caractéristique $V_{Q_i} = \{V_{Q_{i,R_{k_1}}}\}$ et $V_{C_j} = \{V_{C_{j,R_{k_2}}}\}$, où R_k est une région avec $1 \leq k_1, k_2 \leq K$.

$\{V_{Q_{i,R_{k_1}}}\}$ et $\{V_{C_{j,R_{k_2}}}\}$ sont représentés par les sept composantes de la signature, voir la section (§3.6) :

$$V_{Q_{i,R_{k_1}}} = \{V_{Q_{i,R_{k_1,c}}}, 1 \leq k_1 \leq 5\ , 1 \leq c \leq 7\} \tag{3.35}$$
$$V_{C_{j,R_{k_2}}} = \{V_{C_{j,R_{k_2,c}}}, 1 \leq k_2 \leq 5\ , 1 \leq c \leq 7\} \tag{3.36}$$

La distance entre les deux vidéos se traduit par la relation suivante :

$$D(V_Q, V_C) = \sum_{c=1}^{7} \gamma_c FDTW(V_{Q_{i,R_{k_1,c}}}, V_{Q_{j,R_{k_2,c}}}) \tag{3.37}$$

Les γ_v sont des poids d'ajustement utilisés pour favoriser certaines composantes de la signature par rapport aux autres (voir section §3.9)

FDTW représente la distance qui se calcule suivant les 2 étapes suivantes :

Quand K=1, nous avons deux vecteurs caractéristiques $V_{Q_i} = \{V_{Q_{i,R_1}}\}$ et $V_{C_j} = \{V_{C_{j,R_1}}\}$. La distance est calculée en utilisant la FDTW comme dans (§3.8.1).

Nous procédons de la même manière pour construire une matrice D de distances entre paires de régions $k = 1, 2, ..., K$, (typiquement $K = 5$)

$$D = \begin{pmatrix} Fdtw_{1,1} & Fdtw_{1,2} & Fdtw_{1,3} & Fdtw_{1,4} & Fdtw_{1,5} \\ Fdtw_{2,1} & Fdtw_{2,2} & Fdtw_{2,3} & Fdtw_{2,4} & Fdtw_{2,5} \\ Fdtw_{3,1} & Fdtw_{3,2} & Fdtw_{3,3} & Fdtw_{3,4} & Fdtw_{3,5} \\ Fdtw_{4,1} & Fdtw_{4,2} & Fdtw_{4,3} & Fdtw_{4,4} & Fdtw_{4,5} \\ Fdtw_{5,1} & Fdtw_{5,2} & Fdtw_{5,3} & Fdtw_{5,4} & Fdtw_{5,5} \end{pmatrix}$$

Pour obtenir la distance entre les deux vidéos $Q = \{q_i\}$ et $C = \{c_j\}$, l'EMD est appliquée sur la matrice D suivant la formule suivante :

$$EFDTW(Q, C) = \frac{\sum_{i=1}^{K} \sum_{j=1}^{K} f_{i,j} D}{\sum_{i=1}^{K} \sum_{j=1}^{K} f_{i,j}} \tag{3.38}$$

Où $f_{i,j} \geq 0$ représente le travail entre Q et S, minimisant les contraintes suivantes :

$$\sum_{i=1}^{K} f_{i,j} \leq W_{Q_i}, \sum_{j=1}^{K} f_{i,j} \leq W_{C_j} \tag{3.39}$$

$$\sum_{i=1}^{K} \sum_{j=1}^{K} f_{i,j} = min(\sum_{i=1}^{K} W_{Q_i}, \sum_{j=1}^{K} W_{C_j}) \tag{3.40}$$

3.9 Détermination des poids utilisés dans le calcul de la distance

Pour prendre en compte l'importance relative des différentes composantes de la signature, nous devons construire des distances pondérées entre vidéos. Dans ce but, nous allons ajuster les poids pour assurer la meilleure précision possible au niveau sélection des vidéos de la base par rapport aux requêtes. Nous effectuons cet ajustement par apprentissage sur une base de cas connus, renseignés, dont on connait la classe. Plusieurs méthodes d'apprentissage sont proposées dans la littérature.

3.9.1 Position du problème

Soit N le nombre de coefficients de la signature. Nous recherchons les coefficients γ_h qui maximisent l'efficacité du système pour une base de vidéos, la distance entre deux vidéos étant définie par l'équation suivante :

$$D(V_Q, V_C) = \sum_{h=1}^{N} \gamma_h d(V_{Q_{i,R_k}}, V_{C_{j,R_l}}) \tag{3.41}$$

où d est la distance entre les composantes de la signature de même nature pour les deux vidéos Q et C .

Pour évaluer la performance d'une méthode de recherche, nous devons lui attribuer un score sous la forme d'une valeur numérique. La mesure d'évaluation classique des algorithmes de CBVR, la courbe de précision-rappel (cf. chapitre 1, section §3.4), n'est pas adaptée : elle ne permet pas de définir un ordre entre les jeux de paramètres γ_h .

Comme nous l'avons précisé, nous avons choisi la précision moyenne de retrouvaille pour une fenêtre de cinq vidéos comme critère principal d'évaluation des méthodes (cf. chapitre 4, section §4.2.1.2). Nous utilisons donc ce critère pour définir le score de chaque jeu de paramètres γ_k. Nous devons résoudre un problème de maximisation de fonction dans R^N . La fonction à maximiser est donnée par l'équation suivante :

$$f : \begin{pmatrix} R^N \rightarrow [0;1] \\ (\gamma_h)_{h=1..N} \rightarrow \text{précision moyenne} \end{pmatrix}$$

Cette fonction n'est pas continue. En effet, puisque nous comptons le nombre de vidéos correctement sélectionnées, la fonction est à valeur dans un espace dénombrable, de taille N. Par conséquent, un algorithme de descente classique du type gradient conjugué n'est pas adapté. De plus, la fonction pouvant présenter a priori plusieurs maxima locaux, nous ne sommes pas assurés de trouver le maximum global. Une autre approche a donc été utilisée : les algorithmes génétiques. Nous rappelons ci-dessous le principe des algorithmes génétiques.

3.9.2 Les algorithmes génétiques

Les algorithmes génétiques [23] appartiennent à la famille des algorithmes évolutionnistes (un sous-ensemble des méta-heuristiques). Leur but est d'obtenir rapidement une solution approchée à un problème d'optimisation, lorsqu'on ne connait pas de méthode exacte pour le résoudre en un temps raisonnable. Les algorithmes génétiques utilisent la notion de sélection naturelle développée au XIXe siècle par Darwin et l'appliquent à une population de solutions potentielles au problème considéré. On se rapproche par "bonds" successifs d'une solution, comme dans une procédure de séparation et évaluation, à ceci près que ce sont des formules qui sont recherchées et non plus directement des valeurs.

Les algorithmes génétiques sont souvent utilisés pour rechercher les extréma d'une fonction dans un domaine délimité de l'espace, de manière stochastique. Ces algorithmes font évoluer plusieurs solutions (une population de solutions, appelées génomes). A chaque itération (génération), on génère une nouvelle population de solutions obtenues par sélection des individus les plus prometteurs, qui sont recopiés, recombinés deux à deux ou légèrement modifiés (mutés).

Cette procédure permet de converger rapidement vers les extréma locaux tout en se donnant les moyens d'en sortir (grâce aux recombinaisons ou croisements). Les algorithmes sont paramétrés par les probabilités de croisement et de mutation, ainsi que par la manière dont les solutions sont sélectionnées, croisées et mutées, qui conditionnent la qualité des solutions ainsi que la vitesse de convergence.

Les génomes sont composés d'un ensemble de gènes, qui correspondent aux paramètres de la solution. Dans le cas le plus simple, les génomes sont des vecteurs de paramètres. A chaque génome est associé une mesure d'évaluation qui traduit son adaptation au problème. Plus le génome est adapté, plus il aura de chance d'être sélectionné pour former la génération suivante :

L'algorithme le plus simple (voir figure 3.22) consiste à :

- générer N individus à chaque itération (par sélection, recombinaisons et mutations), N étant la taille de la population.
- remplacer les anciens individus par les nouveaux.
- méthodes de sélection : sélection par tournoi [24].
- croisement : on définit une probabilitie de croisement entre les individu.
- mutation : on définit une probabilitie de mutation.

L'algorithme que nous avons utilisé est celui de la création continue (steady state). Son principe est le suivant :

- à chaque itération, l individus ($l < N$) sont générés à partir de la génération précédente.
- les l nouveaux individus sont mélangés avec ceux de la génération précédente.
- les M plus mauvais candidats de la population ($M < N$) résultante sont éliminés.

Contrairement aux algorithmes de descente, les algorithmes génétiques peuvent être facilement parallélisés. En effet, à chaque itération, on évalue indépendamment un certain nombre d'individus générés à partir de la population à l'itération précédente. Ces nouveaux individus peuvent alors être répartis entre différents processeurs pour y être évalués.

Une fois la population évaluée sur les différents processeurs, leur score est donné au processeur maître, qui peut s'en servir pour générer la population suivante. Nous avons donc parallélisé l'algorithme de création continue à l'aide de la librairie MPI [25].

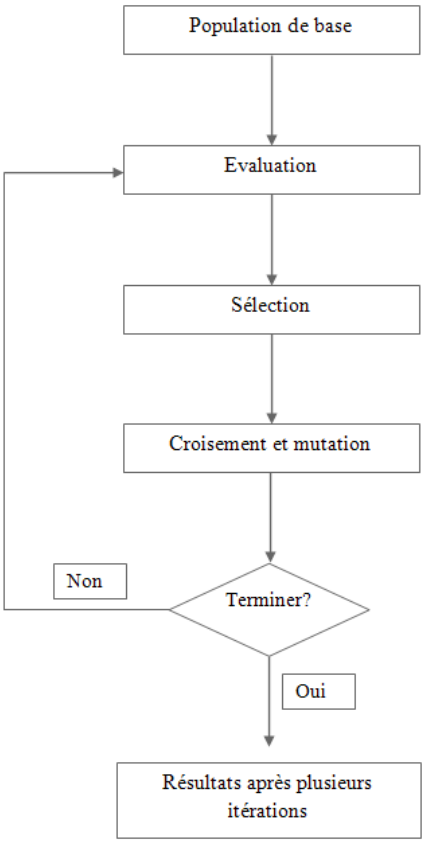


Figure 3.22 — Schéma simple d'un algorithme génétique

3.9.3 Adaptation à la distance étudiée

Dans le cas étudié, les gènes sont les poids affectés à chaque composante de la signature pour le calcul de la distance entre deux séquences d'images, c'est à dire les γ_v, $i = 1...N$; la mesure d'adaptation des génomes est la précision moyenne pour une fenêtre de cinq vidéos. Les paramètres qui ont été utilisés sont donnés par le tableau §3.1.

Tableau 3.1 — Paramètres de l'algorithme génétique utilisé pour la recherche de poids entre les composants de la signature

taille des populations	25
nombre de générations	100
probabilité de mutation	0.06
probabilité de croisement	0.7
méthode de sélection	tournoi
méthode de croisement	pair / impair

Le choix de ces paramètres est justifié dans le chapitre 4, section §4.2.2.

3.9.4 Adaptation des poids aux vidéos de la base

Dans la section précédente, nous avons explicité comment nous adaptons les poids à la distance utilisée pour comparer une vidéo requête avec toutes les vidéos de la base étudiée (voir figure 4.1)

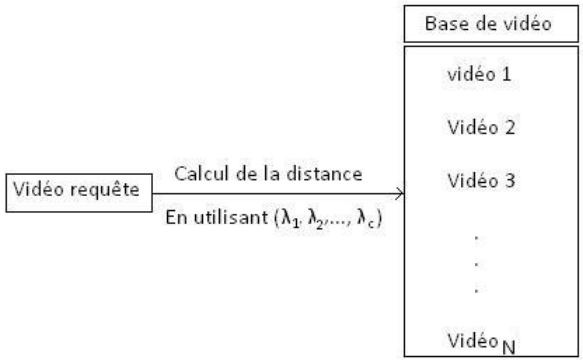


Figure 3.23 — Schéma de calcul de distance entre la vidéo requête et les vidéos de la base en utilisant les même poids

le paramètre c représente le nombre de composante de la signature définies dans l'équation (§3.32) et (§3.35). Ce paramètre varie selon la méthode utilisée (5 composantes ou 7 composantes).

3.10 Conclusion

Dans ce chapitre, nous avons présenté les méthodes que nous proposons pour représenter le contenu des vidéos à partir des données MPEG4, et sélectionner, dans une base de vidéos, celles qui sont les plus proches de la vidéo en requête, par rapport aux signatures choisies.

Pour caractériser nos vidéos, trois méthodes ont été proposées. La première méthode consiste à caractériser globalement la vidéo en utilisant des histogrammes de directions de mouvements. Les deux autres méthodes sont basées sur une segmentation spatio-temporelle et sur le suivi des régions dans la séquence, pour construire une signature décrivant la trajectoire des régions identifiées comme les plus importantes visuellement. A chacune des trois méthodes, nous avons ajouté l'information de résidu pour avoir des signatures plus efficaces et plus riches en terme d'information.

Pour comparer les signatures et permettre la recherche dans des bases de données de vidéos, nous avons proposé trois mesures de distances. Nous avons utilisé dans un premier temps la distance DTW (Dynamic Time Warping) pour le premier type de signatures. Cette distance permet une comparaison complète, fine, entre séquences. Elle est cependant trop coûteuse en temps pour que l'on puisse comparer une requête à l'ensemble des séquences de la base de données avec un temps de réponse raisonnable. Nous avons donc proposé une version approchée, peu coûteuse en temps de calcul, de la distance DTW : la FDTW (Fast Dynamic Time Warping). Pour comparer les signatures issues des méthodes de suivi de régions, nous avons proposé une combinaison de la DTW/FDTW avec la distance EMD (Earth Mover's distance), appelée EFDTW (Extended Fast Dynamic Time Warping). C'est une extension de la DTW dans le domaine multi-dimensionnel où chaque dimension est représentée par la trajectoire d'une région.

Dans le chapitre 4, nous allons présenter les résultats et nous discuterons de l'intérêt de chaque méthode. Nous donnerons auparavant quelques éléments sur le système d'acquisition de vidéos de la chirurgie de la rétine, et présenterons les bases de vidéos que nous avons utilisées.

Bibliographie

[1] W.H. Press et al. Numerical Recipes in C : The Art of Scientific Computing. Cambridge University Press, 1992.

[2] M.N. Do and M. Vetterli. Wavelet-based texture retrieval using generalized gaussian density and kullback-leibler distance. IEEE Trans. Image Processing, 11(2) :146158, february 2002.

[3] http ://fr.wikipedia.org/wiki/H.264

[4] Videolan,a free h264 encoder v1153,200 http ://www.videolan.org/developers/x264.html

[5] Institut nachrichtentechnik heinrich-institut, H.264 reference Software v15.1,2008 http ://iphome.hhi.de/suehring/tml/

[6] JVT Software Page, JM/TML http ://bs.hhi.de/suehring/tml/

[7] Lamard M, Cazuguel G, Roux C, Cochener B : L'utilisation de l'information de mouvement pour la recherche des vidéos médicales par leurs contenus, Journée de Recherche en Imagerie et Technologies de la Santé, Rennes Avril 6-8,, 2011,,0-0,C,A,FR

[8] Yue-Meng Chen, Ivan V. Bajic : Predictive Decoding for Delay eduction in Video Communications. GLOBECOM 2007 : 2053-2057

[9] S. Desmet, B. Deknuydt, L. Van Eycken, and A. Oosterlinck, classified Motion Estimation for video coding, Proceeding of SPIE, Vol, 2182, Image and video processing II, pp

[10] Li Zhao, Quan-li Chen (2007), Implementation of vehicle detection and tracking based on Kalman filter, Electronic Measurement Technology, 30 (2), p. 165-168I.-M

[11] http ://en.wikipedia.org/wiki/Pseudorandom-number-generator

[12] M. J. Swain and D. H. Ballard. Color indexing, International Journal of Computer Vision, 7 :1 1991.

[13] I.-M. Pao and M.-T. Sun, Modeling DCT coefficients for fast video encoding, IEEE Trans. Circuits Syst. Video Technol., vol. 9, no. 4, pp. 608-616, June 1999

[14] W.H. Press et al. Numerical Recipes in C : The Art of Scientific Computing. Cambridge University Press, 1992.

[15] M.K. Varanasi and B. Aazhang. Parametric generalized gaussian density estimation. J. Acoust. Soc. Amer., 86 :14041415, 1989.

[16] Steven M. Kay. Fundamentals of statistical signal processing : estimation theory. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993.

[17] M.N. Do and M. Vetterli. Wavelet-based texture retrieval using generalized gaussian density and kullback-leibler distance. IEEE Trans. Image Processing, 11(2) :146158, february 2002.

[18] W. Hu, D. Xie, Z. Fu, W. Zeng, and S. Maybank, Semantic-based surveillance video retrieval, IEEE Trans Image Process, vol. 16, no. 4,pp. 11681181, 2007.

[19] H. Sakoe S. Chiba. Dynamic Programming Algorithm Optimization for Spoken Word Recognition. IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 26, pages 4349, 1978.

[20] E. Keogh C.A. Ratanamahatana. Exact Indexing of Dynamic Time Warping. Knowledge And Information Systems, vol. 7, no. 3, pages 358386, 2005.

[21] Scaling and time warping in time series querying VLDB '05 Proceedings of the 31st international conference on Very large data bases Pages 649 - 660, 2005.

[22] Y. Rubner, C. Tomasi, and L. J. Guibas. A Metric for Distributions with Applications to Image Databases. Proceedings of the 1998 IEEE International Conference on Computer Vision, Bombay, India, January 1998, pp. 59-66

[23] D. E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning. Boston, 1989.

[24] Tobias Blickle and Lothar Thiele, A Comparison of Selection Schemes Used in Genetic Algorithms,Technical Report 11, Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH) Zurich, Gloriastrasse 35, CH-8092 Zurich, 1995.

[25] http ://www.open-mpi.org/

CHAPITRE 4

INDEXATION ET RECHERCHE DE VIDEO DANS LE DOMAINE COMPRESSÉ : RÉSULTATS

Dans le chapitre 3, nous avons présenté les méthodes que nous proposons pour caractériser les vidéos chirurgicales. Ces méthodes peuvent être utilisées à la fois pour indexer des vidéos, faire des recherches dans des bases de données de vidéos similaires à une vidéo requête, au sens de nos critères, et pour suivre théoriquement des chirurgies en per-opératoire. Nous n'avons cependant pas pu les évaluer dans ce dernier cadre, pour des raisons de temps de calcul d'abord, mais aussi parce qu'il faudra vraisemblablement parvenir à cataloguer les différentes phases des chirurgies, pour faciliter la recherche dans les bases.

L'évaluation de nos méthodes a nécessité la construction de bases de vidéos chirurgicales, et nous commencerons ce chapitre en présentant rapidement le système d'acquisition vidéo utilisé, puis les deux bases que nous avons construites avec le service d'ophtalmologie du CHRU de Brest : une première base relative à la chirurgie de la rétine (pelage de membrane), une deuxième base relative à la chirurgie de la cataracte. Pour comparer nos résultats à des résultats de la littérature, nous avons du utiliser une base non chirurgicale, la base Hollywood, qui sera décrite également.

Nous donnons ensuite les résultats obtenus par chacune des méthodes sur ces trois bases, nous les comparons à des méthodes publiées antérieurement, et nous discutons de ces résultats à la fin du chapitre.

4.1 Bases de données

Pour évaluer nos algorithmes, nous nous sommes intéréssés plus particulièrement à deux bases médicales présentées ci-dessous, concernant la chirurgie ophtalmologique, et plus précisément les chirurgies de la rétine et de la cataracte. Pour mieux comprendre la description de ces deux bases médicales, nous donnons un schéma simplifié de la structure de l'oeil :

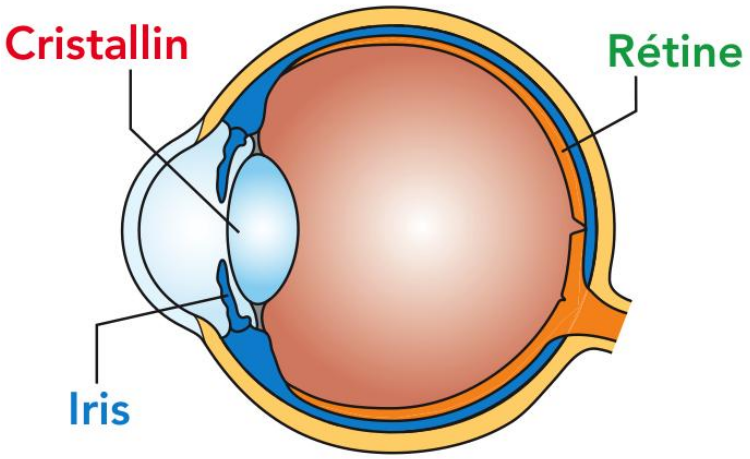


Figure 4.1 — Schéma simplifié de la structure de l'oeil

- **L'iris** est une membrane circulaire et contractile de la face antérieure du globe oculaire. Il constitue la partie colorée visible de l'oeil.
- **Le cristallin** est un petit disque fibreux, transparent et flexible qui permet de focaliser l'image sur la rétine en fonction de la distance.
- **La rétine** est située dans la partie postérieure de l'oeil. C'est la plus interne des membranes de l'oeil, qui est sensible à la lumière et qui transmet l'information au nerf optique.

4.1.1 Description du système d'acquisition de vidéos médicales

La figure 4.5 montre le système d'acquisition développé au sein du service d'ophtalmologie de l'hôpital universitaire de Brest. Avant chaque opération, les données démographiques (âge, sexe, ... etc), et des données contextuelles (présence d'un diabète, une maladie inflammatoire, petite taille de la pupille, ... etc) sont recueillies par une infirmière du bloc opératoire et rendues anonymes.

Nous nous somme intéréssés à deux type de chirurgies, la première est le pelage de membrane et la deuxième est la chirurgie de la cataracte (voir section §4.1.2 et §4.1.3). Les chirurgies ont été effectuées par des chirurgiens différents, dans deux salles d'opération différentes (salle d'opération 1, salle d'opération 2). Une infirmière a été en charge de l'enregistrement vidéo des interventions chirurgicales. Dans la salle d'opération 1, les vidéos ont été enregistrées avec un appareil CCD-IRIS (Sony, Tokyo, Japon) et un magnétoscope DSR-20MDP (Sony, Tokyo, Japon). Dans la salle d'opération 2, les vidéos ont été enregistrées avec une caméra 3CCD incorporée dans le microscope et un enregistreur vidéo MediCap USB200 (MediCapture, Philadelphie, USA).

Les vidéos enregistrées au sein de la salle d'opération 1 sont stockées au format MPEG2, avec les paramètres qui donnent les meilleures qualités, et en format DV (Digital Video) dans la salle d'opération 2. A la fin de chaque opération, le chirurgien remplit un formulaire mentionnant le type de la chirurgie, l'implant utilisé pour la chirurgie de la cataracte, les réglages de l'appareil utilisé et les complications constatées au cours de la chirurgie (cf. Annexe B).

Le protocole d'étude est conforme aux principes de la déclaration d'Helsinki. Le comité de protection des personnes du CHU de Brest a approuvé le protocole de l'étude, et les vidéos ayant été anonymisées, une dispense de consentement a été accordée.

Figure 4.2 — Système d'acquisition numérique de chirurgies de la rétine

4.1.2 Base de chirurgies de pelage de membrane rétinienne

Une membrane épi-rétinienne est composée de cellules provenant de la rétine et qui se sont étalées à sa surface en un tissu très fin. Au début de son évolution, la membrane épi-rétinienne entraîne peu de gêne visuelle. Mais elle peut s'épaissir ou devenir opaque et être alors responsable d'une **baisse de l'acuité visuelle**, qui s'installe le plus souvent progressivement et qui devient sévère avec le temps. Ce symptôme est souvent la première plainte du patient et peut être à lui seul une indication d'intervention chirurgicale.

Sur le plan épidémiologique, l'apparition d'une membrane épirétinienne est une pathologie relativement commune chez la personne âgée. Dans 20% des cas, les MER (membrane épi-rétinienne) sont secondaires à diverses pathologies oculaires, telles que des déchirures rétiniennes, des occlusions vasculaires, des inflammations intra-oculaires. Dans 80% des autres cas, on ne retrouve pas de pathologies oculaires associées, elle est alors appelée membrane épimaculaire.

La chirurgie des membranes épimaculaires permet d'obtenir de bons résultats fonction-

nels avec une amélioration de l'acuité visuelle et une disparition du syndrome maculaire en postopératoire dans la majorité des cas. De plus, la possibilité de réaliser cette chirurgie par voie transconjonctivale a diminué l'invalidité postopératoire, sans supprimer cependant les complications qui restent habituelles de la vitréctomie.

Le pelage de membrane se compose de trois étapes :

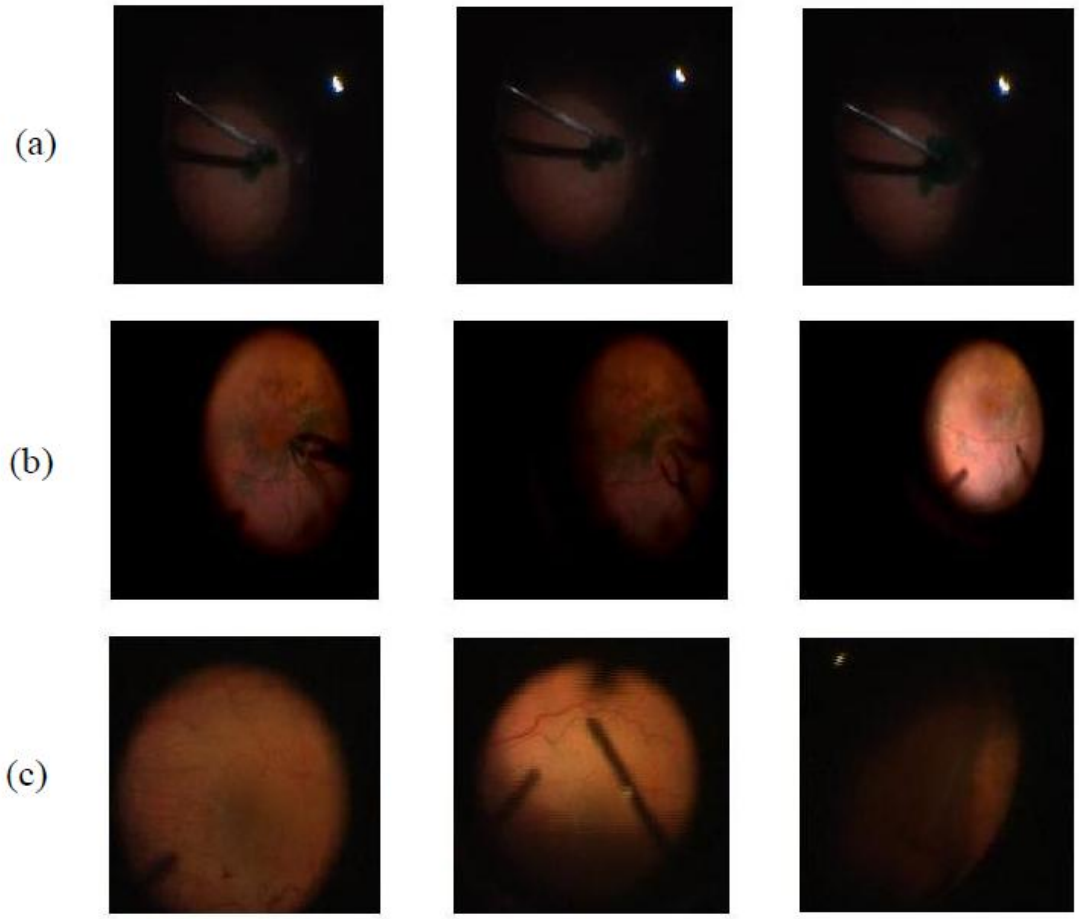


Figure 4.3 — Image de la chirurgie de pelage de membrane : (a) étape d'injection, (b) étape de pelage, (c) étape de vitréctomie

- **Injection** : cette étape consiste à injecter un colorant tel que le vert d'indocyanine pour faciliter l'ablation de la membrane épirétinienne.
- **Pelage** : dans un second temps, le chirugien procède à l'ablation de la membrane épirétinienne, tissu très fin étalé sur la surface de la rétine.
- **Vitréctomie** : après injection du colorant et ablation de la membrane épirétinienne, cette étape consiste à retirer le corps vitré qui est la substance transparente gélatineuse qui remplit l'oeil.

La base de vidéos de pelage de membranes, développée spécifiquement pour l'étude, contient 23 cas. Les films de la chirurgie ont une taille moyenne de 621s (écart type de 299s) avec des images de résolution 720x576.

Un chirurgien ophtalmologiste a segmenté chaque vidéo en trois classes : injection, pelage et vitréctomie, correspondant aux étapes de l'opération de pelage de membrane. (3 classes pour chaque film = 69 vidéos).

Tableau 4.1 — Base de pelage de membrane

Nombre de la classe	Classe	Nombre de vidéo
1	Injection	23
2	Pelage	23
3	Vitréctomie	23

4.1.3 Base de chirurgies de la cataracte

La cataracte est une opacification totale ou partielle du cristallin (lentille naturelle de l'oeil), qui entraîne une **baisse de la vision**. Elle peut apparaître dans un ou dans les deux yeux. La gêne visuelle varie selon l'intensité et l'emplacement des opacités.

Le cristallin normal est transparent. Il laisse passer la lumière jusqu'à la partie postérieure de l'oeil (sur la rétine) pour que les images soient clairement vues. Si certaines parties du cristallin deviennent troubles (opaques), la lumière ne peut pas les traverser et, de ce fait, la vision baisse. Première cause de cécité au monde (environ 48 % des cas de cécité), la cataracte est un important problème de santé publique, particulièrement dans les pays en voie de développement. La cataracte est principalement liée au vieillissement : elle est donc souvent inévitable. En France, elle touche plus de 60 % des personnes de plus de 85 ans.

Le seul traitement efficace de la cataracte est **la chirurgie**. L'opération a été inventée par le chirurgien Franco au XVIe siècle. L'intervention aujourd'hui consiste à enlever le cristallin opaque, et le remplacer par un cristallin artificiel (implant intra-oculaire) qui prend place dans l'enveloppe du cristallin (appelée capsule) laissée en place pendant l'intervention. Cette intervention est actuellement très au point, et se fait classiquement sous anesthésie de contact ou locale.

L'intervention se fait le plus souvent en ambulatoire, c'est-à-dire sans hospitalisation, ou alors avec une hospitalisation très courte, selon les cas. La complication postopératoire la plus fréquente de l'intervention chirurgicale est la cataracte secondaire qui peut apparaître quelques jours à quelques années après l'intervention. Elle correspond à une opacification de la capsule. Cette opacification se traite par capsulotomie, le plus souvent au laser Nd-YAG. Des impacts focalisés sur la capsule vont la déchirer et rendre immédiatement une vue normale. Il arrive également que l'un des points de suture sur la cornée ne soit plus parfaitement étanche. Le chirurgien observe alors le signe de Seidel, qui traduit la fuite d'humeur aqueuse à travers la perforation. La prise en charge doit être rapide et adaptée, l'oeil étant exposé à un grand risque septique.

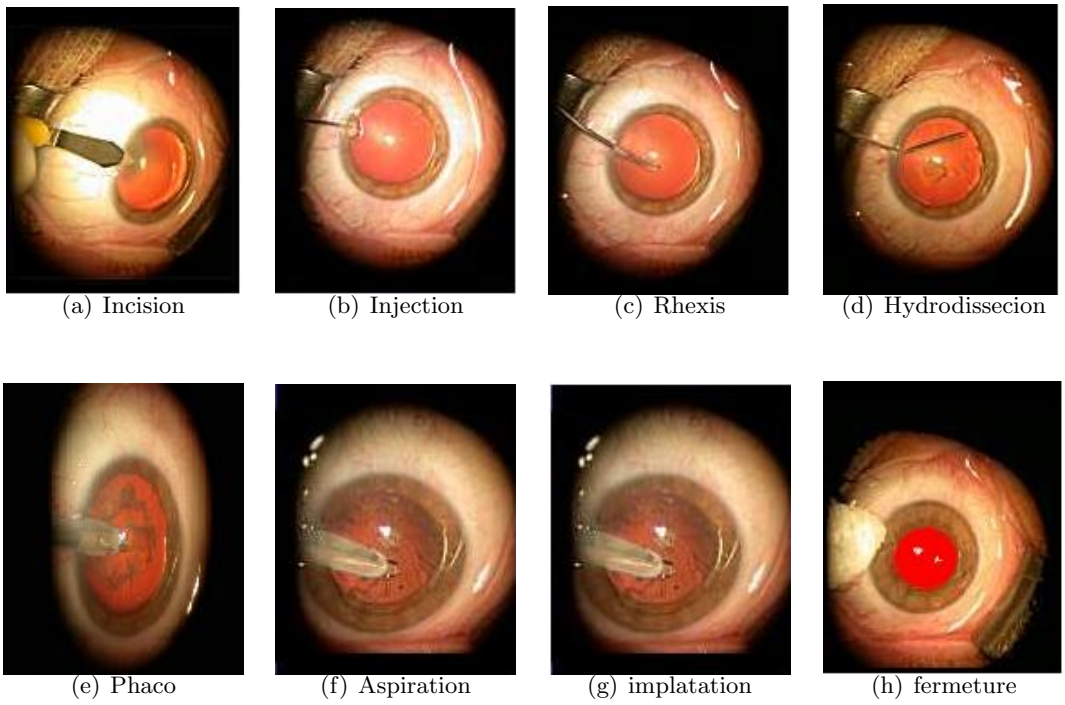
Dans les pays du tiers monde, l'intervention préférentielle (pour des raisons de coût) reste l'extraction intra-capsulaire du cristallin, où l'enveloppe (la capsule) est retirée en même temps que ce dernier. Les résultats sont moins bons que l'extraction extra-capsulaire.

La base de vidéos de la cataracte, réalisée aussi pour notre l'étude, contient 250 cas. Les films de la chirurgie ont une taille moyenne de 17 min et 16 s (écart type de 10 min et 25 s) avec une définition des images de 720x576 pixels. Un chirurgien ophtalmologiste a décrit chaque opération (voir Annexe B) et il a affecté 11 classes à chaque vidéo : Incision, Injection, Rhexis, Hydrodissection ... etc), correspondant aux étapes de l'opération de la catarcte, comme résultats nous avons 1,638 séquences vidéo.

– **Incision :** sous anesthésie locale, ou simplement par collyre anesthésique, l'intervention débute par l'incision cornéenne au couteau pour accéder à la chambre antérieure de

l'oeil.

- **Injection de visqueux :** par cette incision, on remplit la chambre antérieure d'un produit visqueux (en général du hyaluronate de sodium) afin de garder une bonne profondeur de cette chambre antérieure car l'humeur aqueuse, trop fluide, a tendance à s'échapper par l'incision.
- **Rhexis :** une aiguille permet d'inciser la capsule antérieure du cristallin (capsulorhexis curviligne) ainsi le noyau du cristallin apparait directement et va être accessible.
- **Hydrodissection :** il faut maintenant que ce noyau pivote dans son sac grâce à l'injection d'humeur aqueuse artificielle. Ceci se fait à l'aide d'une canule qui glisse entre le sac et le noyau.
- **Phako :** une sonde à ultrason va pouvoir maintenant creuser le noyau du cristallin dans différents secteurs (ici en croix).
- **Aspiration ou Epi noyau :** autour du noyau, persistent des reliquats du cristallin qu'il faut aspirer à l'aide d'une sonde d'aspiration.
- **Mise en place ou implantation :** un implant souple, pliable est introduit plié sur lui même à travers l'incision puis mis en place.
- **Fermeture :** elle peut être auto étanche mais il peut parfois être nécessaire de mettre un point de suture.
- **RetraitVisqueux :** en fin de l'opération, le visqueux est remplacé par de l'eau stérile.
- **Divers :** toute sorte d'action qui ne constitue pas une classe chirurgicale (insérer le ciseau, ...etc)
- **Rien :** la classe ou rien ne se passe au cours de l'opération.



(a) Incision (b) Injection (c) Rhexis (d) Hydrodissecion

(e) Phaco (f) Aspiration (g) implatation (h) fermeture

Figure 4.4 — Images des étapes de la chirurgie de la cataracte

Tableau 4.2 — Base de chirurgie de la cataracte

Numéro	Classe	Nombre de vidéo
1	Incision	105
2	Injection	114
3	Rhexis	115
4	Hydrodissection	118
5	Phako	126
6	Fermeture	111
7	Implantation	123
8	Epinoyau	118
9	RetraitVisqueux	123
10	Divers	28
11	Rien	466

4.1.4 Base HOLLYWOOD

Cette base de données a été utilisé pour montrer la généricité des méthodes proposées. Elle contient 1837 vidéos extraites de différents films de hollywood [1], issue d'une collaboration entre le laboratoire INRIA et l'IRISA en France (HOLLYWOOD2 human action dataset).

Les vidéos qui la composent sont classifiées en fonction de leur nature en terme d'action humaine, 12 classes sont affectées aux vidéos :

- répondre au téléphone
- conduire
- manger
- chute d'une personne
- sortir de la voiture
- agiter à la main
- câlin
- embrasser
- courir
- s'asseoir
- s'allonger
- se lever

La base est divisée en deux parties approximativement identiques ; la première représente la base d'apprentissage et contient 823 vidéos, la deuxième est la base de test et elle en contient 884. Les images ont différentes résolutions (640x352 pixels, 576x312 pixels et 548x226 pixels). La présense de l'une des 12 classes est indiquée dans chacune des vidéos (1 si la classe apparait dans la vidéo, 0 sinon).

Les statistiques de la base sont données dans le tableau suivant :

Tableau 4.3 — La base Hollywood

Numéro	**Classe**	**Nombre de vidéo**	**Définition**
1	Répondre au téléphone	130	640x352
2	Conduire	178	576x312
3	Manger	73	548x226
4	Chutte d'une personne	124	640x352
5	Sortir de la voiture	108	576x312
6	Agiter à la main	77	640x352
7	Câlin	130	548x226
8	Embrasser	217	576x312
9	Courir	276	640x352
10	S'asseoir	212	548x226
11	S'allonger	61	576x312
12	Se lever	278	640x352

Figure 4.5 — Des images extraites de la base Hollywood

4.2 Méthodologie

Les méthodes ont été développées dans le cadre de l'aide aux gestes de la chirurgie en ophtalmologie. Nous précisons ici les objectifs visés dans ce cadre et les critères d'évaluation qui en découlent. Les mêmes critères d'évaluation sont appliqués aux trois bases de données.

Comme nous l'avons mentionné en introduction de la thèse, notre travail ne s'inscrit pas dans le cadre d'une assistance robotique, mais dans l'assistance au chirurgien lors de son opération afin de le prévenir de risques (cas des alertes) ou pour prodiguer des conseils sur les décisions pour assurer le bon déroulement de la chirurgie.

4.2.1 Critères d'evaluation des 3 méthodes proposées

4.2.1.1 Evaluation pour chaque base de vidéos

Chaque base de vidéos est divisée en 2 parties de taille (approximativement) égales (base de test et base d'apprentissage). Pour la base pelage de membrane, au vu du peu de vidéos qu'elle contient (69 vidéos) comparée aux deux autres bases de vidéos, nous avons utilisé la validation croisée : la base est divisée en 2 parties, on réalise l'apprentissage en laissant à chaque fois une des parties de côté pour le valider.

4.2.1.2 Précision moyenne à 5

Le mode d'évaluation est adapté à l'objectif poursuivi. Il s'agit de retrouver dans la base de cas disponibles, uniquement les cas les plus probablement voisins de la requête pour prévoir des alertes ou des préconisations sur tel ou tel geste chirugical. Les médecins jugent que les cinq premières vidéos sélectionnées par le système sont suffisantes, en particulier pour des raisons de temps et au vu des résultats fournis par le système. Le système est donc paramétré afin de maximiser la pertinence des cinq premiers cas proposés. Pour évaluer les performances des méthodes proposées, nous calculons donc le pourcentage moyen de vidéos pertinentes parmi les cinq premiers résultats proposés lors d'une requête. Pour chaque base de vidéos, nous procédons ainsi :

- chaque vidéo de la base de test est placée en requête à tour de rôle.
- la distance entre la vidéo requête et chaque vidéo de la base de test est calculée (selon la méthode utilisé (DTW, FDTW, EFDTW) (cf. chapite 3, section §3.7)).
- les vidéos de la base de test sont classées par ordre de distance croissante.
- une précision moyenne à 5 est calculée, c'est-à-dire le pourcentage de vidéos ayant la même classe que la vidéo requête parmi les 5 vidéos les plus proches.

Dans les résultats (§4.3), les courbes de précision-rappel sont fournies à titre indicatif, mais nous ne les utilisons pas pour comparer les résultats des différentes méthodes car elles sont peu significatives pour les médecins et notre application.

4.2.1.3 Précision moyenne

Pour évaluer la généricité des méthodes proposées, nous les avons également évaluées sur la base Hollywood. Nous comparons nos méthodes avec le travail de M.Marszalek et al. [5], dans lequel les résultats sont exprimés par la précision moyenne et non pas en précision moyenne à 5.

Pour pouvoir comparer nos méthodes, nous avons donc exprimé nos résultats également en précision moyenne. Nous avons paramétré notre apprentissage pour nos trois méthodes, de manière à maximiser cette précision moyenne (nos trois méthodes avec résidu et avec apprentissage).

La précision moyenne se détermine de la manière suivante :

- chaque vidéo de la base de test est placée en requête à tour de rôle.
- la distance entre la vidéo requête et chaque vidéo de la base de test est calculée (selon la méthodes utilisé (DTW, FDTW, EFDTW) (cf. chapite 3, section §3.7)).
- les vidéos de la base de test sont classées par ordre de distance croissante.
- une précision moyenne est calculée (MAP : Maximized Average Precision) par l'équation suivante :

$$MAP = \frac{\sum_{l=1}^{n}(pertinence(l)/l)}{\text{nombre des vidéos pertinentes dans la base de vidéos}} \tag{4.1}$$

Où $pertinence(l)$ représente le nombre des vidéos pertinentes déja trouvées, et l le rang de la vidéo dans la base

Pour comprendre l'intérèt de cette mesure, nous présentons deux exemples. Supposons qu'une base de vidéos comporte sept vidéos parmi lesquelles trois sont pertinentes (vidéos ayant la même classe que la vidéo requête)

Exemple 1 :

Tableau 4.4 — Précision moyenne (exemple 1)

Classement	Pertinence	précision
T1	Oui	1
T2	Oui	2/2
T3	Non	-
T4	Oui	3/4
T5	Non	-
T6	Non	-
T7	Non	-

Ce qui donne :

$$MAP = \frac{1}{3}(1 + \frac{2}{2} + \frac{3}{4}) = 0.91 \tag{4.2}$$

Exemple 2 :

Tableau 4.5 — Précision moyenne (exemple 2)

Classement	Pertinence	précision
T1	Non	-
T2	Oui	1/2
T3	Non	-
T4	Oui	2/4
T5	Non	-
T6	Non	-
T7	Oui	3/7

Pour cet exemple, la précision moyenne est :

$$MAP = \frac{1}{3}(\frac{1}{2} + \frac{2}{4} + \frac{3}{7}) = 0.47 \tag{4.3}$$

Ce nombre MAP mesure bien la qualité du classement, grâce à un score évoluant entre 0 et 1.

4.2.2 Choix des paramètres utilisés pour les trois méthodes

Remarque :

les paramètres utilisés pour la compression de vidéo sont ceux de la norme de compression MPEG. L'ensemble des paramètres utilisés pour les trois méthodes proposées a été basé sur une étude bibliographique et des ajustements empiriques. En effet, le temps de calcul pour un seul jeu de paramètres, avec apprentissage, pour une méthode appliquée sur une seule base de données nécessite environ 1 mois de calcul (la taille des bases de données est donnée dans §4.1). C'est la raison pour laquelle nous n'avons pas pu tester plusieurs jeux de paramètres.

Les paramètres utilisés nous ont amenés à des résultats très encourageants (voir le paragraphe §4.3). Cependant, il aurait été plus satisfaisant de pouvoir tester d'autres jeux de paramètres. Nous évoquerons quelques possibilités pour pallier ce problème dans la conclusion.

4.2.2.1 Choix de la taille du GOP

Le choix du nombre d'images entre deux images I ou ce qu'on appelle GOP (Group Of Pictures) (cf. chapitre 2) du codage des vidéos MPEG est un des paramètrage de l'algorithme de compression. MPEG utilise le plus souvent une taille de GOP de 15 à 18 images [2]. Nous avons choisi 15 images par Gop pour avoir plus d'images I et une meilleure description de la vidéo.

4.2.2.2 Nombre de classes choisies pour classifier le mouvement (cf. chapitre 3, section §3.2)

Le choix du nombre de classes de l'histogramme de classifications des vecteurs de mouvement $K = 13$ a été établi après une étude de la littérature. Le même type d'histogramme a déjà été utilisé avec succès pour d'autres application de classification de mouvement [3] [4].

4.2.2.3 Algorithme de segmentation par croissance de région à partir d'un germe (cf. chapitre 3, section §3.3.2)

Les deux paramètres utilisés sont ajustés empiriquement pour obtenir une bonne segmentation. $T_{Max\_offset} = 8$: ce paramètre représente la variance maximale qu'on peut avoir dans une région, donc la variance d'une région est limitée à 8. Ceci est réalisé pour vérifier que si la variance est trop grande au sein de la région (ce qu'on ne souhaite pas puisqu'on cherche des régions avec des blocs à déplacements cohérents) on ne regroupe pas des blocs par erreur.

$T_{Mov\_region} = 4$: seuil de regroupement de deux régions lié aussi à la variance entre deux régions qu'on souhaite fusionner.

4.2.2.4 Mesure de similitude entre deux images I pour la sélection des GOPs (cf. chapitre 3, section §3.4.1)

Pour détecter les images I les plus représentatives, le paramètre de seuil $T_{Macrobloc}$ est défini empiriquement égal à 0.7. Une valeur trop petite implique une selection plus élevée et une valeur trop grande implique une perte de l'information entre les images I sélectionnées. Pour éviter ceci, ainsi que le problème de l'utilisation d'histogrammes (il est possible de ne pas détecter la différence si les deux images concernées ont un histogramme similaire mais un contenu différent), nous avons choisi de décomposer les images en macrobloc de taille $L = 8$ pour empêcher l'erreur de se propager sur toute l'image.

4.2.2.5 Les paramètres de l'apprentissage avec algorithme génétique

Nous remarquons que pour les paramètres d'optimisation par algorithme génétique donnés dans le tableau §3.1 (cf. chapitre 3, section §3.9.3) :

- Il vaut mieux privilégier le nombre de générations par rapport à la taille des populations. Cependant, dans certains cas, la population arrête rapidement d'évoluer. Nous avons donc prévu d'arrêter l'algorithme quand les solutions ne s'améliorent plus.
- Le fait d'avoir une probabilité de croisement importante (70%) permet de sortir facilement des maxima locaux.
- La méthode de sélection par tournoi consiste à sélectionner deux individus avec une probabilité proportionnelle à leur adaptation (le score exprimé en précision moyenne à 5) et à conserver le meilleur des deux, c'est à dire celui qui maximise la précision.

4.3 Résultats

Nous allons donner les résultats obtenus par les trois méthodes présentées précédemment avec et sans optimisation. Les résultats sont présentés sous forme de courbes précision-rappel, pour chaque base de données. Les scores de précision moyenne sont récapitulés dans le tableau 4.6.

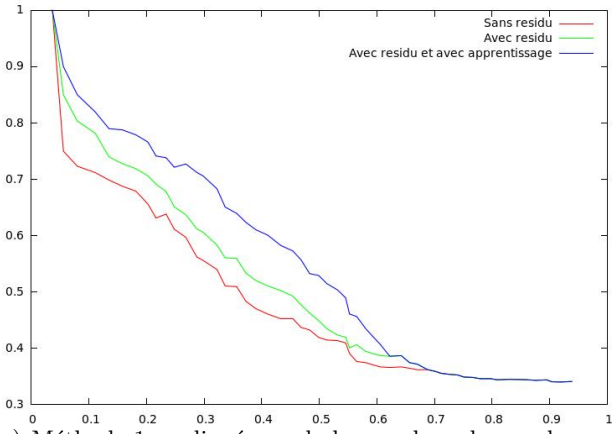
Tableau 4.6 — Précision moyenne pour une fenêtre de cinq vidéos (précision moyenne à 5) pour les 3 méthodes proposées

Bases de données	**Pelage de membrane**			**Hollywood**			**Cataracte**		
Méthodes	sans résidu	avec résidu	avec résidu + appren-tissage	sans résidu	avec résidu	avec résidu + appren-tissage	sans résidu	avec résidu	avec résidu + appren-tissage
Méthode 1 (§3.2)	69,3%	74%	**79,69%**	65,3%	71,33%	**75,69%**	67,8%	70%	**72,69%**
Méthode 2 (§3.3)	62,8%	69,66%	73%	60%	67,33%	72,5%	59,33%	67%	71,4%
Méthode 3 (§3.4)	64,3%	67,66%	75,2%	65,6%	69%	74,3%	63,3%	67,5%	72,3%

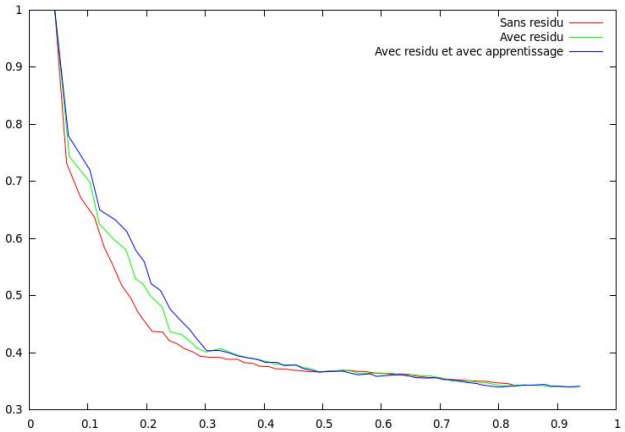
D'après le tableau 4.6, nous remarquons que les meilleures performances sont obtenues en utilisant l'approche basée sur l'orientation et l'intensité de mouvement en exploitant la séquence complète, ce qui est logique. En effet elle fournit une analyse et des caractéristiques de la vidéo beaucoup plus riche que les méthodes qui n'utilisent que les images I ou les GOPs.

Ces résultats montrent aussi l'intérêt de l'utilisation de l'information de résidu qui apporte un gain très important dans la précision moyenne à 5 (jusqu'à 10% dans certains cas). Une autre amélioration proposée dans ce travail, est celle des poids adaptés globalement à chaque base de test de chaque base de vidéos. Le jeu de paramètres qui maximise la précision moyenne à 5 a été retenu.

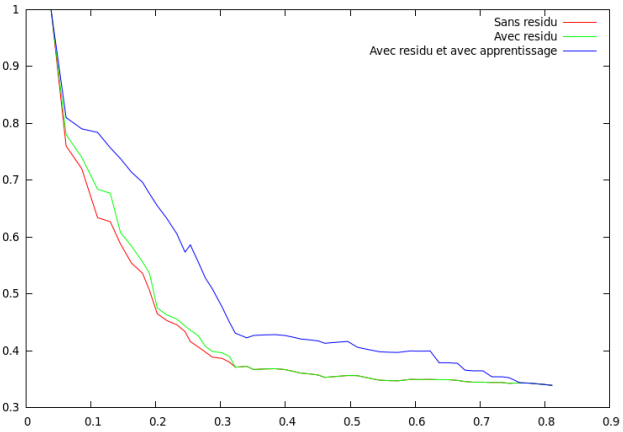
Nous donnons dans les figures 4.6, 4.7 et 4.8, des courbes de précision-rappel (cf. chapitre 1, section §1.4) montrant l'influence de l'information de résidu ainsi que celle de l'apprentissage sur la précision, en utilisant les trois méthodes pour les trois bases de vidéos utilisées.



(a) Méthode 1 appliquée sur la base pelage de membrane

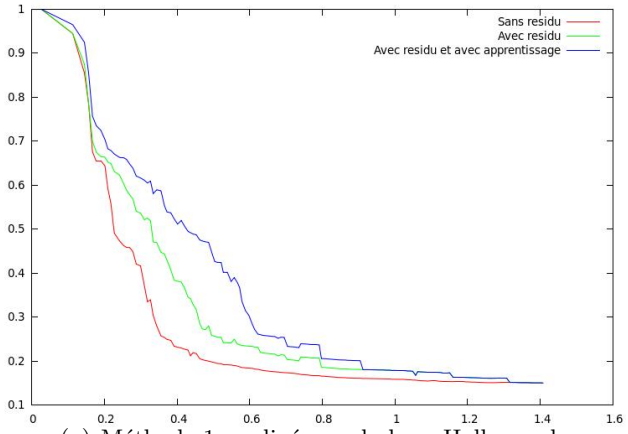


(b) Méthode 2 appliquée sur la base pelage de membrane

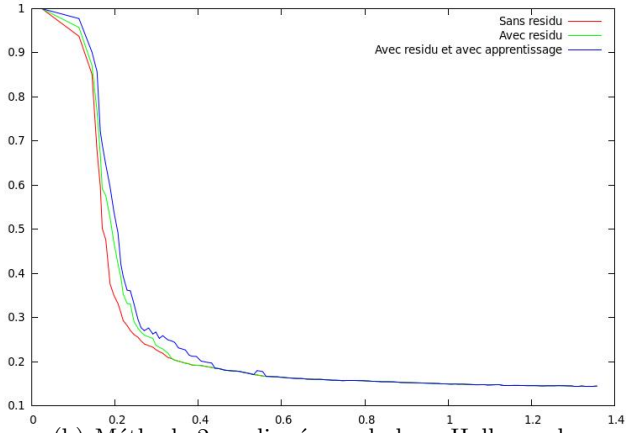


(c) Méthode 3 appliquée sur la base pelage de membrane

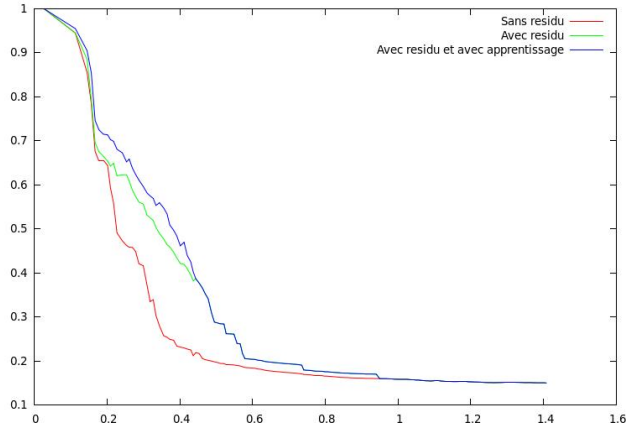
Figure 4.6 — Influence de la combinaison de l'information de résidu avec la signature et l'apprentissage sur la précision moyenne à 5, en utilisant les 3 méthodes proposées pour la base de pelage de membrane



(a) Méthode 1 appliqée sur la base Hollywood

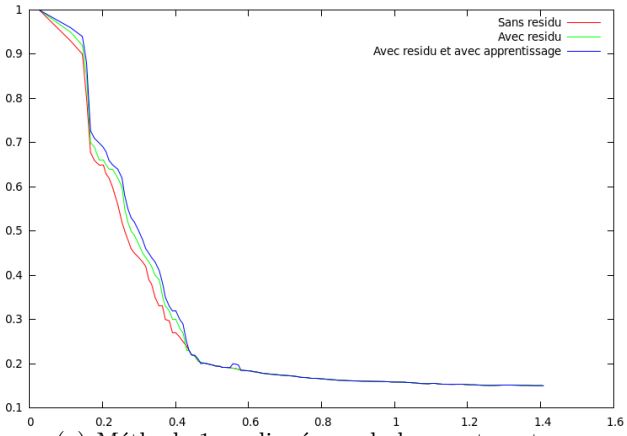


(b) Méthode 2 appliquée sur la base Hollywood

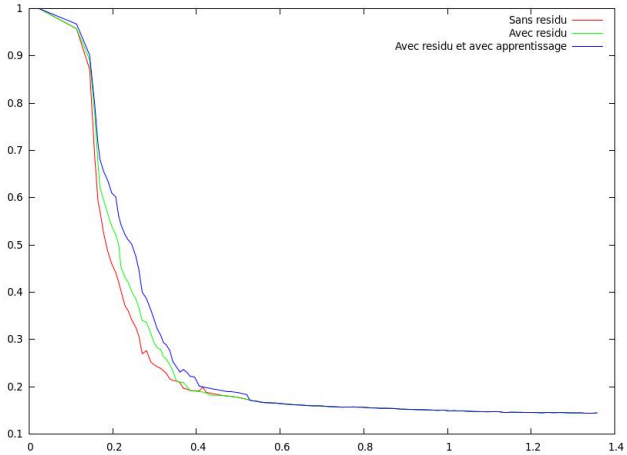


(c) Méthode 3 appliquée sur la base Hollywood

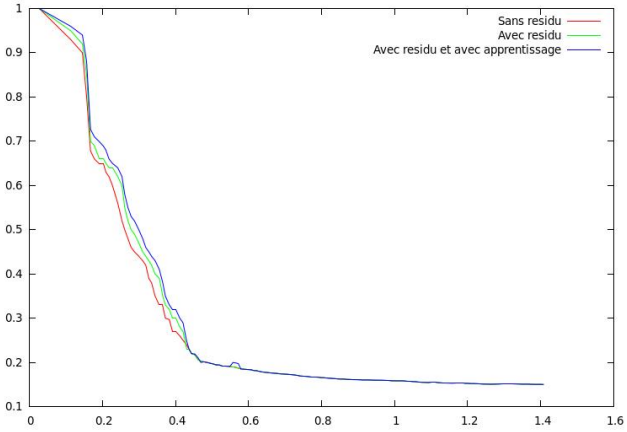
Figure 4.7 — Influence de la combinaison de l'information de résidu avec la signature et l'apprentissage sur la précision moyenne à 5, en utilisant les 3 méthodes proposées pour la base de Hollywood



(a) Méthode 1 appliquée sur la base cataractes



(b) Méthode 2 appliquée sur la base cataractes



(c) Méthode 3 appliquée sur la base cataractes

Figure 4.8 — Influence de la combinaison de l'information de résidu avec la signature et l'apprentissage sur la précision moyenne à 5, en utilisant les 3 méthodes proposées pour la base cataractes

Dans les figures 4.6, 4.7 et 4.8, les courbes précision-rappel confirmes nos remarques concernant nos résultats éxprimés en précision moyenne à 5 (voir tableau 4.6). Les meilleures performances sont obtenues en utilisant l'approche basée sur l'orientation et l'intensité de mouvement. Nous voyons sur les courbes, l'influence de l'information de résidu et celle des poids adaptés, et le fait qu'elles apportent un gain significatif.

À titre indicatif, nous illustrons dans le tableau 4.7, les résultats de la précision moyenne pour une fenêtre d'une vidéo (une fenêtre d'une vidéo et non pas cinq vidéos) pour les 3 méthodes proposées :

Tableau 4.7 — Précision moyenne pour une fenêtre d'une vidéo pour les 3 méthodes proposées

Bases de données	**Pelage de membrane**			**Hollywood**			**Cataracte**		
Méthodes	sans résidu	avec résidu	avec résidu + apprentissage	sans résidu	avec résidu	avec résidu + apprentissage	sans résidu	avec résidu	avec résidu + apprentissage
Méthode 1 (§3.2)	75%	85%	**90%**	94,5%	96,5%	**97,5%**	93%	95%	**96,5%**
Méthode 2 (§3.3)	73,18%	74,33%	78%	93,75%	95,75%	97%	95,75%	95,75%	96%
Méthode 3 (§3.4)	76,01%	78,14%	81,01%	92,5%	94,5%	95,5%	92,5%	94,5%	95,5%

D'après le tableau 4.7, nous remarquons que pratiquement, une vidéo pertinente est classée en première position dans plus de 90% des cas pour la base pelage de membrane jusqu'à 97% des cas pour la base Hollywood.

4.3.1 Temps de calcul

Dans ce travail nous devons concevoir des approches avec un temps de calcul admissible compatible avec le travail peropératoire. La méthode 1 (cf. chapitre 3, section §3.2) est lente malgré son efficacité. Nous avons améliorer le temps de calcul en utilisant uniquement les images I ((cf. chapitre 3, section §3.3)) et la méthode basée sur la selection des Gops ((cf. chapitre 3, section §3.4)), pour permette l'utilisation du système en temps réel. Nous donnons dans le tableau 4.8, un exemple de résultats sur le temps de calcul, pour rechercher les cinq vidéos les plus proches d'une vidéo requête de 9min, dans chacune des bases. Tous les calculs ont été effectués par un processeur AMD Athlon 64-bit cadencé à 2 GHz et 8G de mémoire.

Tableau 4.8 — Temps de calcul moyen de recherche d'une vidéo de 9 minutes dans une base de données

Dataset	**Pelage de membrane**	**Hollywood**	**Cataracte**
Méthode 1	temps 1 : 14 min 33s	temps 1 : 12 min 25s	temps 1 : 13 min 25s
	temps 2 : 3 min 25s	temps 2 : 6 min 55s	temps 2 : 5 min 15s
	Total :17 min 58s	**Total : 19 min 20s**	**Total : 18 min 40s**
Méthode 2	temps 1 : 7 min 03s	temps 1 : 6 min 55s	temps 1 : 7 min 07s
	temps 2 : 1 min 10s	temps 2 : 6 min 20s	temps 2 : 5 min 35s
	Total : 8 min 13s	**Total : 13 min 15s**	**Total : 12 min 42s**
Méthode 3	temps 1 : 6 min 38s	temps 1 : 6 min 03s	temps 1 : 6 min 49s
	temps 2 : 1 min 03s	temps 2 : 5 min 20s	temps 2 : 5 min 53s
	Total : 7 min 41s	**Total : 11 min 23s**	**Total : 12 min 43s**

temps 1 : le temps nécessaire pour le calcul du vecteur caractéristique pour une vidéo de 9 minutes
temps 2 : le temps nécessaire pour calculer la distance avec chaque vidéo dans la base

Nous voyons que le temps de recherche de l'approche basée sur les images I (méthode 2) ou celle basée sur la sélection des GOPs (méthode 3) est nettement plus court que celui de la méthode basée sur l'extraction de l'orientation et l'intensité du mouvement exploitant la séquence complète. Avec les méthodes 2 et 3, le système donne les meilleurs résultats du point de vue temps, mais avec une précision moins bonne que le méthode 1. Ces résultats montrent que l'amélioration de la précision moyenne à 5 apportée par la méthode 1 se fait au détriment de la rapidité de la réponse aux requêtes.

4.4 Comparaison de nos méthodes aux travaux antérieurs

Remarque :

la comparaison de nos méthodes avec celles développés dans le laboratoire IRISA/INRIA Rennes France [5], n'est effectuée que sur la base Hollywood. Les résultats dans [5] sont exprimés en précision moyenne (§4.2.1.3) et non pas en précision moyenne à 5 (§4.2.1.2). Nous avons refait l'apprentissage des poids pour les trois méthodes, pour maximiser la précision moyenne (nos trois méthodes avec résidu et apprentissage) pour la comparaison.

Dans [5], les auteurs examinent plusieurs caractérisations des vidéos. Ils utilisent d'abord les descripteurs SIFT (Scale-invariant feature transform). Il s'agit d'extraire les informations numériques dérivées de l'analyse locale d'une image/vidéo et qui caractérisent le contenu visuel de la façon la plus indépendante possible de l'échelle (zoom et résolution du capteur, du cadrage, de l'angle d'observation et de la luminosité). La deuxième méthode proposée dans [5], est la description de vidéos par histogramme de gradient orienté (HOG). Une troisième méthode est de représenter le contenu des vidéos par le descripteur HOF qui représente un histogramme des orientations du flot optique, décrit par Laptev et al [6] [7].

Les meilleurs résultats présentés dans [5], sont obtenus en combinant les trois descripteurs (voir table 4.8). Les résultats obtenus sur la base Hollywood dans [5] sont donnés dans le tableau suivant :

Tableau 4.9 — Précision moyenne des trois méthodes données dans [5]

Méthodes	SIFT	HOG HOF	SIFT HOG HOF
Hollywood	20%	32,4%	32,6%

Dans le tableau ci-dessous nous illustrons les résultats obtenus avec nos méthodes et celle basée sur les trois descripteurs donnée dans [5] :

Tableau 4.10 — Précision moyenne en utilisant nos trois méthodes et celles proposées dans [5]

Méthodes	Méthode 1 (résidu + Apprentissage)	Méthode 2 (résidu + Apprentissage)	Méthode 3 (résidu + Apprentissage)	SIFT HOG HOF [5]
Hollywood	**36,9%**	26%	**34,2%**	32,6%

Dans ce tableau 4.10, nous voyons la différence entre les méthodes qu'on a proposées et celles étudiés dans [5]. Un gain significatif jusqu'à 4,3 % est obtenu en utilisant la méthode basée sur l'orientation et l'intensité du mouvement, exploitant toutes les images de la séquence, et 1.7% en utilisant la méthode basée sur la selection des GOPs (cf. chapitre 3, section §3.4). Cela peut être expliqué par le fait que l'utilisation des méthodes de segmentation et de suivi, sur un nombre suffisant d'images (qui est le cas avec la méthode basée sur les GOPs et non pas celle basée sur les images I), est mieux adaptée que les méthodes basées sur les descripteur SIFT, HOG et HOF décrit dans [5], sur la base de hollywood. Au niveau

temps de calcul, les auteurs [5] ne donnent pas d'indication.

4.5 Discussion

L'efficacité des méthodes proposées, mesurée par la précision moyenne pour une fenêtre de cinq vidéos, est intéressante : elle atteint les 79% (4 vidéos sont similaires à la vidéo requête) pour la base de pelage de membrane, 75,69% pour la base de Hollywood (3 à 4 vidéos sont similaires en moyenne à la vidéo requête) et 72,69% pour la base de la cataracte (3 à 4 vidéos sont similaires à la vidéo requête). Les meilleurs résultats sont obtenus en utilisant la méthode 1 en exploitant la séquence complète ce qui est logique, du fait qu'elle fournit une analyse et des caractérisations de vidéo beaucoup plus riches que celles basées sur le suivi des régions homogènes entre les images I, ou avec sélection des GOPs. Bien que la base de chirurgies de pelage de membrane ne contienne que peu de vidéos, avec seulement 3 classes représentant les étapes de la chirurgie (69 séquences), les bases Hollywood et chirurgies de la cataracte contiennent beaucoup plus de vidéos (12 classes et 1837 vidéos pour la première, 11 classes et 1638 vidéos pour la seconde). Les résultats que nous obtenons peuvent donc être considérés comme bien significatifs pour les performances obtenues par les trois méthodes.

Les courbes de précision rappel et les valeurs de la précision moyenne à 5, montrent que l'information du résidu et l'apprentissage global des poids, combinés avec les méthodes utilisées pour la génération des signatures, ont augmenté significativement la précision moyenne à 5. Les poids optimaux trouvés par le processus d'apprentissage fournissent dans tous les cas une meilleure précision que les méthodes sans apprentissage. Le choix des types de signatures (entre celles basées sur l'orientation et l'intensité de mouvement sur la séquence complète, le suivi de régions entre les images I et le suivi des régions dans les GOPs sélectionnés) dépend du type de l'application. Par exemple, pour une application ou le temps de calcul est primordial, ce sont les méthodes basées sur le suivi de régions qui sont plutôt à retenir.

Du point de vue des temps de calcul, les approches basées sur le suivi de régions entre les images I ou entre images des GOPs sélectionnés, sont presque 2 fois plus rapides que celles utilisant l'intensité et l'orientation de mouvement. Pour la base pelage de membranes, l'ensemble peut même être du temps réel car le calcul des signatures peut être obtenu à la fin de l'enregistrement vidéo (moins de 7mn pour la vidéo enregistrée compressée, et 1mn de recherche dans la base, mais qui est petite). C'est la recherche dans les bases qui va prendre de plus en plus de temps quand la dimension de la base augmentera. La méthode basée sur le suivi de régions dans les GOPs sélectionnés (méthode 3) est un bon compromis entre la meilleure précision moyenne de la méthode 1 et le temps de calcul de cette méthode 3. Le problème du temps de calcul peut être résolu par l'utilisation d'ordinateurs puissants ou bien d'une grille de calcul pour permettre une utilisation en temps réel dans le cas des grandes bases de données.

La comparaison de nos résultats avec les résultats obtenus dans [5] pour la base de Hollywood donne un avantage à la méthode basée sur l'intensité et l'orientation de mouvement et celle basée sur la sélection des GOPs. Outre l'amélioration de la précision moyenne, nous utilisons le domaine compressé pour extraire les paramètres. Nous évitons ainsi la décompression totale de la vidéo, comparé aux méthodes de description (SIFT, HOG et HOF) donnée dans [5], basées sur l'extraction de paramètres du flot optique, ce qui se traduit par un gain supplémentaire au niveau temps de calcul. La baisse au niveau de la précision moyenne constatée pour la méthode basée sur le suivi de régions entre les images I, peut être justifier par le peu de données qu'on utilise pour extraire la signature ; nous utilisons une image toutes les 15 images, ce qui peut être pénalisant en terme de richesse de la signature.

Pour les trois méthodes proposées, la baisse de la précision moyenne à 5 en augmentant la taille de la base de vidéos peut facilement s'expliquer. Le nombre de vidéos utilisées dans le cas de la base Hollywood est 1,707 vidéos, soit 24 fois le nombre de vidéos de la base pelage de membrane (69 vidéos). La base Hollywood a de plus 12 classes alors que la base pelage n'en a que 3. Ceci peut se traduire par une plus grande dispersion des données qui explique des performances moins bonnes avec les grandes bases.

4.6 Conclusion

Ce chapitre a présenté le système d'acquisition de vidéo médicale installé au sein du service d'ophtalmologie de brest, ainsi que les bases de vidéos utilisées pour cette étude. L'ensemble des résultats obtenus sur ces bases sont aussi exposés. Les meilleures performances sont obtenues par l'approche basée sur l'orientation et l'intensité de mouvement, cependant cette méthode est lente, malgré son efficacité comme nous avons pu le voir. Nous avons trouvé un compromis entre la précision moyenne et le temps de calcul en ne conservant que les parties de la vidéo où il se passe de l'action. Pour évaluer les méthodes proposées, nous avons utilisé la précision moyenne à 5 tout au long de notre étude. Cette mesure fournit des résultats significatifs pour les médecins. Pour pouvoir comparer nos résultats obtenus avec ceux obtenus dans [5], nous avons utilisé la précision moyenne.

Pour conclure, c'est la méthode basée sur la sélection des GOPs qu'il faudra privilégier dans des applications où le temps représente le facteur principal. Il faudrait maintenant disposer d'autres bases pour pouvoir généraliser éventuellement cette conclusion.

Bibliographie

[1] http ://www.irisa.fr/vista/actions/hollywood2/

[2] http ://en.wikipedia.org/wiki/MPEG-1

[3] K. Schoeffmann, M. Lux, M. Taschwer, and L. B ?sz ?rmenyi, Visualization of video motion in context of video browsing, Multimedia and Expo, 2009. ICME 2009. IEEE International Conference on, August 2009, pp. 658 ?661.

[4] Manfred del Fabro, Laszlo Boszormenyi, Video Scene Detection Based On Recurring Motion Patterns, Second International Conferences on Advances in Multimedia, 2010

[5] M. Marszaek, I. Laptev, and C. Schmid, Actions in context, in Proc IEEE Conf Computer Vision Pattern Recognition, 2009, pp.

[6] I. Laptev, B. Caputo, C. Schuldt, and T. Lindeberg. Local velocity-adapted motion events for spatio-temporal recognition. Computer Vision and Image Understan- ding, 108(3) :207229, 2007.

[7] I. Laptev and T. Lindeberg. Local descriptors for spatio-temporal recognition. Computer and Information Science, 3667 :91103, 2006.

[8] http ://en.wikipedia.org/wiki/InformationretrievalMeanaverageprecision

Conclusion

DANS ce mémoire, nous avons abordé la problématique de l'utilisation peropératoire des bases de données médicales multimédia, pour l'aide à la décision. Dans cet objectif, nous nous sommes intéressés aux méthodes permettant de caractériser automatiquement les vidéos, pour mettre en oeuvre des systèmes d'indexation et de recherche d'information par le contenu. Les solutions proposées ont été évaluées dans le cadre d'applications médicales. Le domaine cible était l'ophtalmologie et plus particulièrement les chirurgies de la rétine et de la cataracte. Nous résumons ci-dessous le contenu du mémoire, puis nous explorerons les perspectives de recherche mises en valeur par ce travail de thèse.

Dans le premier chapitre, nous avons présenté les principes des systèmes de recherche d'information, en mettant l'accent sur les systèmes de recherche par le contenu. Nous avons rappelé quels étaient leurs principaux éléments et l'architecture de base de tels systèmes. À la fin de ce chapitre, nous avons montré l'intérêt de l'utilisation des données vidéos dans le domaine compressé, afin d'éclairer les choix qui ont conduit à notre travail de recherche. Dans le second chapitre, nous avons donné en détail le principe de fonctionnement des codeurs vidéo, et plus particulièrement le codeur H.264/AVC sur lequel nous nous sommes appuyés pour créer des signatures de vidéos pour la recherche de vidéos par le contenu (CBVR : Content Based Video Retrieval). Nous avons expliqué chacune des étapes de codage à l'aide d'un schéma basique de codage de cette norme. L'objectif était d'exposer brièvement le cadre global de la compression vidéo et de faire ressortir les éléments permettant de développer des signatures associées aux principales méthodes de compression et/ou dans les différentes étapes de ces méthodes.

Le troisième chapitre a été consacré à la construction des signatures numériques à partir du domaine compressé. Trois méthodes ont été proposées. La première méthode consiste à caractériser globalement la vidéo en utilisant les histogrammes des directions de mouvement. La deuxième méthode est basée sur une segmentation spatio-temporelle et sur le suivi des régions entre deux images I, pour construire une signature décrivant la trajectoire des régions identifiées comme les plus importantes d'un point de vue aire. La troisième méthode est une amélioration de la deuxième méthode, qui permet de pallier la perte d'information de la méthode 2, qui ne prend en compte que les images I. Nous avons construit un résumé de la vidéo basé sur la sélection des GOPs et nous construisons des signatures de la même manière qu'avec la deuxième méthode mais en prenant toutes les images des GOPs sélectionnés. À chacune des trois méthodes, nous avons ajouté l'information de résidu pour avoir des signatures plus efficaces et plus riches en terme d'information.

Après la spécification des méthodes de génération des signatures, une mesure de similitude est définie pour chaque méthode. Nous avons présenté dans un premier temps l'algorithme classique "alignement dynamique temporel", ou "Dynamic Time Warping (DTW)", qui permet d'obtenir efficacement l'ensemble des déformations de coût minimal. Ce type d'algorithme nous permet de prendre en compte les différences de durée des phases des interventions chirurgicales. L'algorithme FDTW que nous utilisons pour comparer les signatures

issues de la première méthode est un algorithme rapide dérivé de l'algorithme DTW. Dans un second temps, nous détaillons la distance EMD (Earth Mover's Distance) qui nous a conduit à la distance EFDTW, en la combinant avec l'algorithme FDTW. Nous l'utilisons pour comparer les signatures des deuxième et troisième méthodes basées sur la trajectoire des régions.

Le chapitre 4 est consacré à la présentation des résultats et à la méthodologie d'évaluation. C'est aussi dans ce chapitre que nous décrivons le système d'aquisition mis en place au sein du service d'ophtalmologie de l'hôpital universitaire de Brest, ainsi que les différentes bases de vidéos utilisées : deux bases de vidéos de chirurgies en ophtalmologie - une base de chirurgies de pelage de membrane et une base de chirurgies de la cataracte - et une base de vidéos souvent utilisée pour des comparaisons entre travaux sur l'indexation et la CBVR, la base Hollywood2.

Pour améliorer les résultats de retrouvaille, un processus d'optimisation des poids des éléments des signatures a été introduit dans le calcul des distances entre la vidéo requête et les vidéos de la base. Ce processus est basé sur une technique d'apprentissage qui utilise les algorithmes génétiques. La précision moyenne, pour une fenêtre de cinq vidéos, obtenue par ces méthodes atteint les 79% (4 vidéos sont similaires à la vidéo requête) pour la base pelage de membrane, 75,69% pour la base Hollywood (3 à 4 vidéos sont similaires à la vidéo requête) et 72,69% pour la base cataracte (3 à 4 vidéos sont similaires à la vidéo requête). Ces résultats sont comparés aux résultats d'une méthodes basée sur une combinaison des descripteurs (SIFT, HOG et HOF). Les méthodes 1 et 3 que nous proposons donnent de meilleurs résultats en terme de précision moyenne (gain de 1,6 à 4,3%).

Le choix de la méthode à préconiser est guidé par le compromis précision-temps de calcul : c'est la méthode basée sur la selection des GOPs et le suivi des régions dans ces GOPs pour obtenir une trajectoire des régions qui nous semble être la plus intéressante. Cependant, quelle que soit la méthode utilisée, la précision moyenne obtenue et la robustesse sont suffisamment importantes pour que toutes les méthodes proposées puissent être utilisées pour la mise en place de systèmes d'aide au geste chirugical.

Des améliorations sont d'ores et déjà envisageables :

- introduire des méthodes d'optimisation pour le calcul du couple distance/signature de parallélisation de code (algorithmes) pour permette l'utilisation en temps réel.
- coupler l'information du mouvement avec d'autres paramètres pouvant être extraits du domaine compréssé (texture (DCT), taille des blocs, ...etc) pour produire des signatures plus riches en termes d'information.
- utiliser la dernière norme de compression HEVC ou H.265, une norme qui offre une efficacité deux fois supérieure à la norme précédente H.264 et dans un temps plus court grâce au calcul parallèle.
- fusionner l'information numérique et textuelle : prise en compte dans toutes les vidéos, des informations contextuelles disponibles (âge, sexe, etc.), éventuellement des connaissances a priori sur l'opération étudiée, etc.
- appliquer les approches sur d'autres bases de vidéos

Nos signatures actuelles nous permettent de rechercher dans les bases de données des phases chirurgicales segmentées dans une vidéo. Mais nous avons montré que ces signatures pouvaient se calculer au moins dans la durée de la vidéo. Il faudra les adapter à un suivi dynamique, par exemple en considérant des segments de trajectoires, ou des groupes d'histogrammes. Avec des séquences plus courtes à considérer, le temps de calcul ne devrait pas poser de problème. Les distances utilisées devraient aussi bien se prêter à un suivi dynamique,

mais il faudra envisager des recherches plus rapides, prenant en compte les informations déjà déterminées sur le déroulement de la chirurgie pour mettre en place des recherches optimisées dans les bases de données, en temps réel.

Il faudra aussi réfléchir au développement d'interfaces homme-machine, pour donner aux praticiens l'information sous une forme ergonomique en situation interventionnelle, et leur permettre d'interagir avec les systèmes, de manière à avoir plus de précisions sur les informations désirées, et d'effectuer des validations qualitatives de ce type de système d'aide à la décision.

En conclusion, nous espérons que ce travail contribuera à la mise en place de systèmes d'aide à la décision basés sur la recherche par le contenu, pour tirer profit de la masse de données et d'informations médicales qui sont numérisées et archivées tous les jours. Ces systèmes devraient faciliter la pratique clinique quotidienne des médecins dans un avenir proche, pour le plus grand bénéfice des patients et des praticiens.

ANNEXE

A Publications

A.1 Revue internationale avec comité de lecture

- Droueche Z, Quellec G, Lamard M, Cazuguel G, Cochener B, Roux C : Computer-Aided Retinal Surgery using Data from the Video Compressed Stream, *IEEE Transactions on Biomedical Engineering*, February 2012.(En révision)
- Droueche Z, Quellec G, Lamard M, Cazuguel G, Cochener B, Roux C : Content-Based MPEG-4 Video Stream Retrieval for Computer-Aided Eye Surgery, *IEEE Transaction on Information Technology in Biomedicine*,(En cours)

A.2 Conférences avec actes et comité de lecture

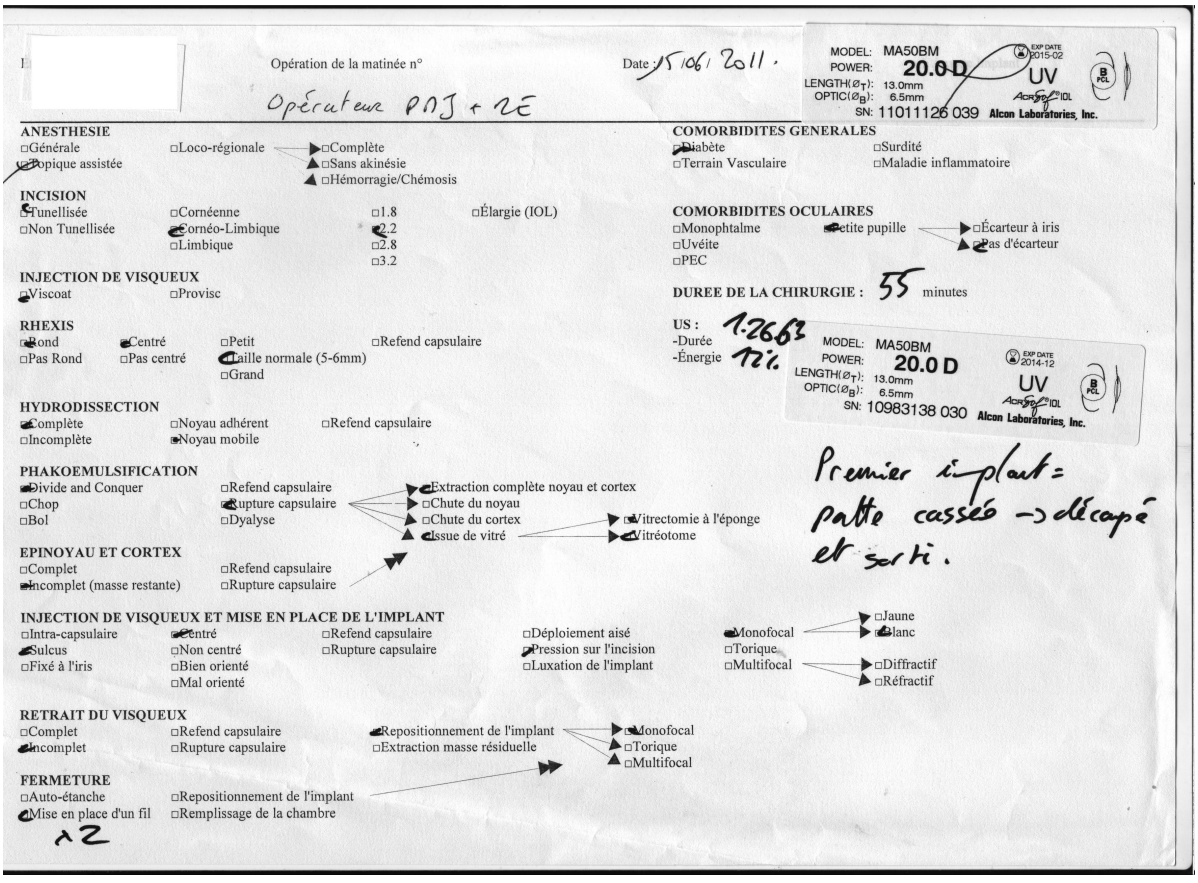
- Droueche Z, Quellec G, Lamard M, Cazuguel G, Roux C, Cochener B : Motion-based Video Retrieval with Application to Computer-Assisted Retinal Surgery, International Conference of the *IEEE Engineering in Medicine and Biology Society*, San diego USA, 2012,,0-0,C,A,GB
- Lamard M ,Cochener B, Quellec G, Cazuguel G, Roux C, Droueche Z : Computer-aided Retinal Surgery Using Video Data Compression, *ARVO Annual Meeting*, Florida, 6-9 May 2012.
- Droueche Z, Lamard M, Cazuguel G, Quellec G, Roux C, Cochener B : Content-based medical video retrieval based on region motion trajectories, *5th European Conference of the International Federation for Medical and Biological Engineering*, Budapest, 14-18 September,, 2011,,0-0,C,A,E
- Quellec G, Lamard M, Cazuguel G, Droueche Z, Roux C, Cochener B : Real-Time Retrieval of Similar Videos with Application to Computer-Aided Retinal Surgery, *International Conference of the IEEE Engineering in Medicine and Biology Society*, 2011,,0-0,C,A,GB
- Droueche Z, Lamard M, Cazuguel G, Quellec G, Roux C, Cochener B : Fouille de séquence d'images médicales. Application en chirurgie mini-invasive augmentée, *Journée des doctorants et post-doctorants en biologie santé en Bretagne*, 20 juin, 2011,,0-0,A,A,FR
- Droueche Z, Lamard M, Cazuguel G, Roux C, Cochener B : L'utilisation de l'information de mouvement pour la recherche des vidéos médicales par leurs contenus, *Journée de Recherche en Imagerie et Technologies de la Santé*, Rennes Avril 6-8,, 2011,,0-0,C,A,FR
- Quellec G, Lamard M, Cazuguel G, Droueche Z, Roux C, Cochener B : Recherche en temps réel de séquences vidéo similaires par le contenu. *TAIMA*, Tunisie 2011,,0-

0,C,A,FR

ANNEXE

B Renseignement de la base de la cataracte

Nous donnons dans cette annexe des exemples de description de chirurgie de la cataracte effectués par différentes médecin au sein de CHU de Brest.



I Opération de la matinée n° Date : 15/06/2011.

Opérateur PMJ + ZE

MODEL: MA50BM
POWER: 20.0 D
LENGTH(ØT): 13.0mm
OPTIC(ØB): 6.5mm
SN: 11011126 039
EXP DATE 2015-02
UV
Alcon Laboratories, Inc.

ANESTHESIE
□Générale □Loco-régionale → □Complète □Sans akinésie □Hémorragie/Chémosis
Topique assistée

COMORBIDITES GENERALES
□Diabète □Surdité
□Terrain Vasculaire □Maladie inflammatoire

INCISION
□Tunellisée □Cornéenne □1.8 □Élargie (IOL)
□Non Tunellisée Cornéo-Limbique 2.2
□Limbique □2.8
□3.2

COMORBIDITES OCULAIRES
□Monophtalme Petite pupille → □Écarteur à iris □Pas d'écarteur
□Uvéite
□PEC

INJECTION DE VISQUEUX
Viscoat □Provisc

DUREE DE LA CHIRURGIE : 55 minutes

RHEXIS
□Rond Centré □Petit □Refend capsulaire
□Pas Rond □Pas centré Taille normale (5-6mm)
□Grand

US : 1.26 63
-Durée 72
-Énergie

MODEL: MA50BM
POWER: 20.0 D
LENGTH(ØT): 13.0mm
OPTIC(ØB): 6.5mm
SN: 10983138 030
EXP DATE 2014-12
UV
Alcon Laboratories, Inc.

HYDRODISSECTION
Complète □Noyau adhérent □Refend capsulaire
□Incomplète □Noyau mobile

Premier implant = patte cassée → décapé et sorti.

PHAKOEMULSIFICATION
Divide and Conquer □Refend capsulaire → Extraction complète noyau et cortex
□Chop Rupture capsulaire → □Chute du noyau
□Bol □Dyalyse → □Chute du cortex → Vitrectomie à l'éponge
→ Issue de vitré → Vitréotome

EPINOYAU ET CORTEX
□Complet □Refend capsulaire
Incomplet (masse restante) □Rupture capsulaire

INJECTION DE VISQUEUX ET MISE EN PLACE DE L'IMPLANT
□Intra-capsulaire Centré □Refend capsulaire □Déploiement aisé Monofocal → □Jaune Blanc
Sulcus □Non centré □Rupture capsulaire Pression sur l'incision □Torique
□Fixé à l'iris □Bien orienté □Luxation de l'implant □Multifocal → □Diffractif □Réfractif
□Mal orienté

RETRAIT DU VISQUEUX
□Complet □Refend capsulaire Repositionnement de l'implant → Monofocal
Incomplet □Rupture capsulaire □Extraction masse résiduelle → □Torique □Multifocal

FERMETURE
□Auto-étanche □Repositionnement de l'implant
Mise en place d'un fil □Remplissage de la chambre

AZ

Figure B.1 — Description de l'opération de la cataracte établie par le Dr. Josselin

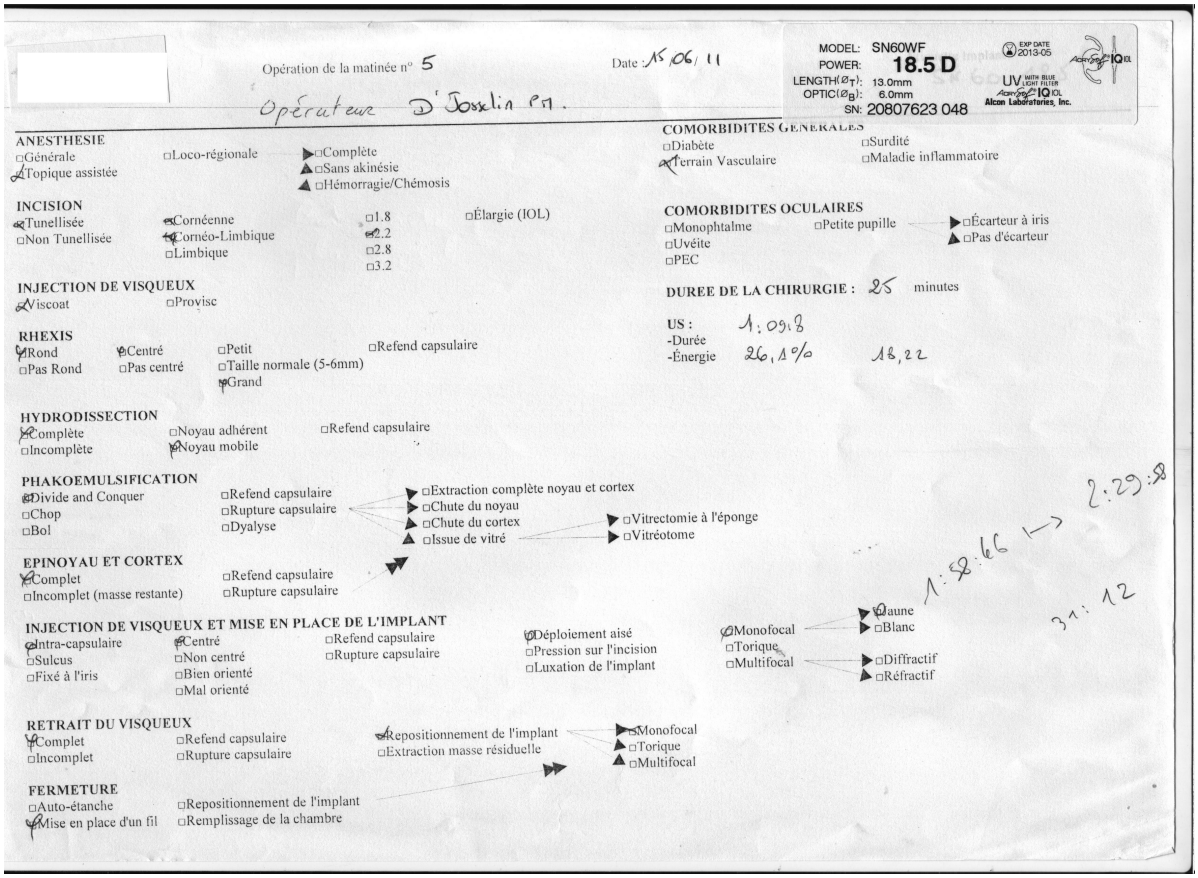


Figure B.2 — Description de l'opération de la cataracte établie par le Dr. Josselin

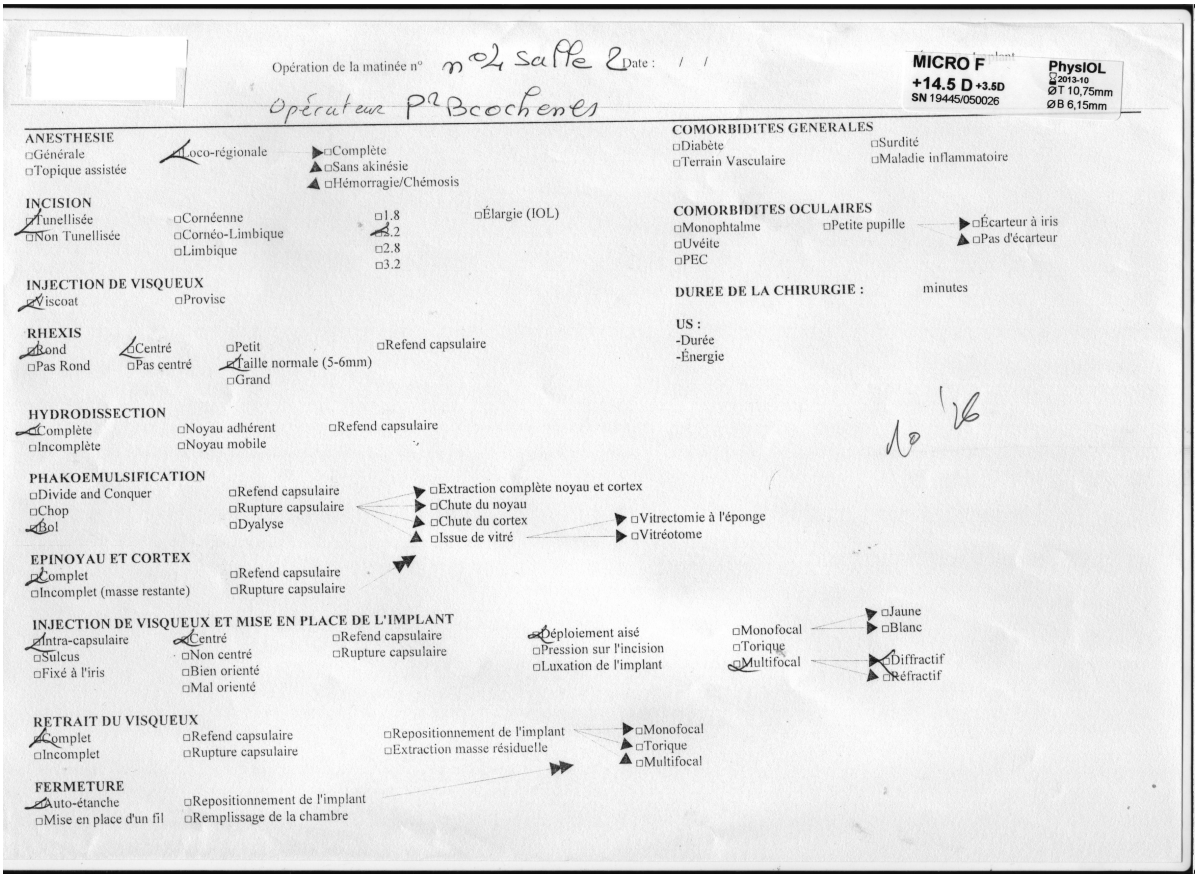


Figure B.3 — Description de l'opération de la cataracte établie par le Dr. Cochener

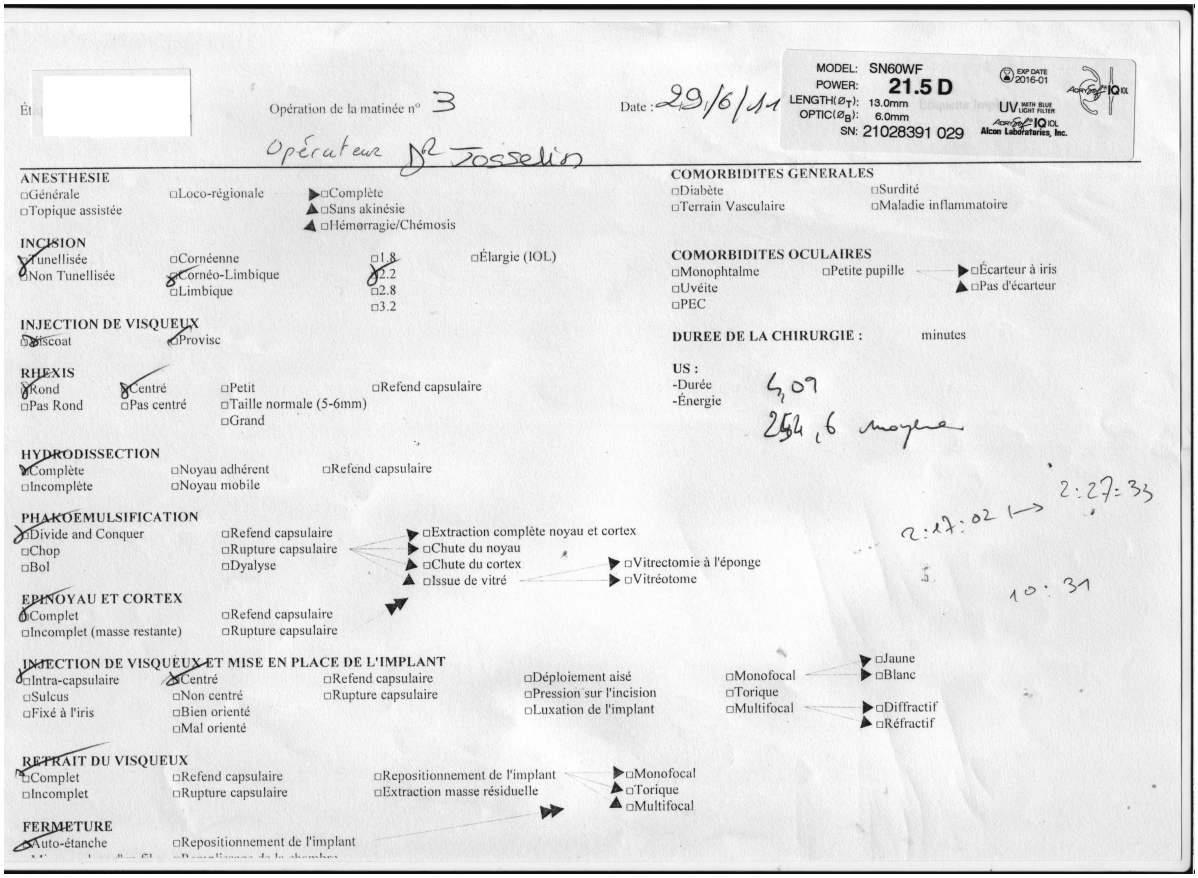


Figure B.4 — Description de l'opération de la cataracte établie par le Dr. Josselin

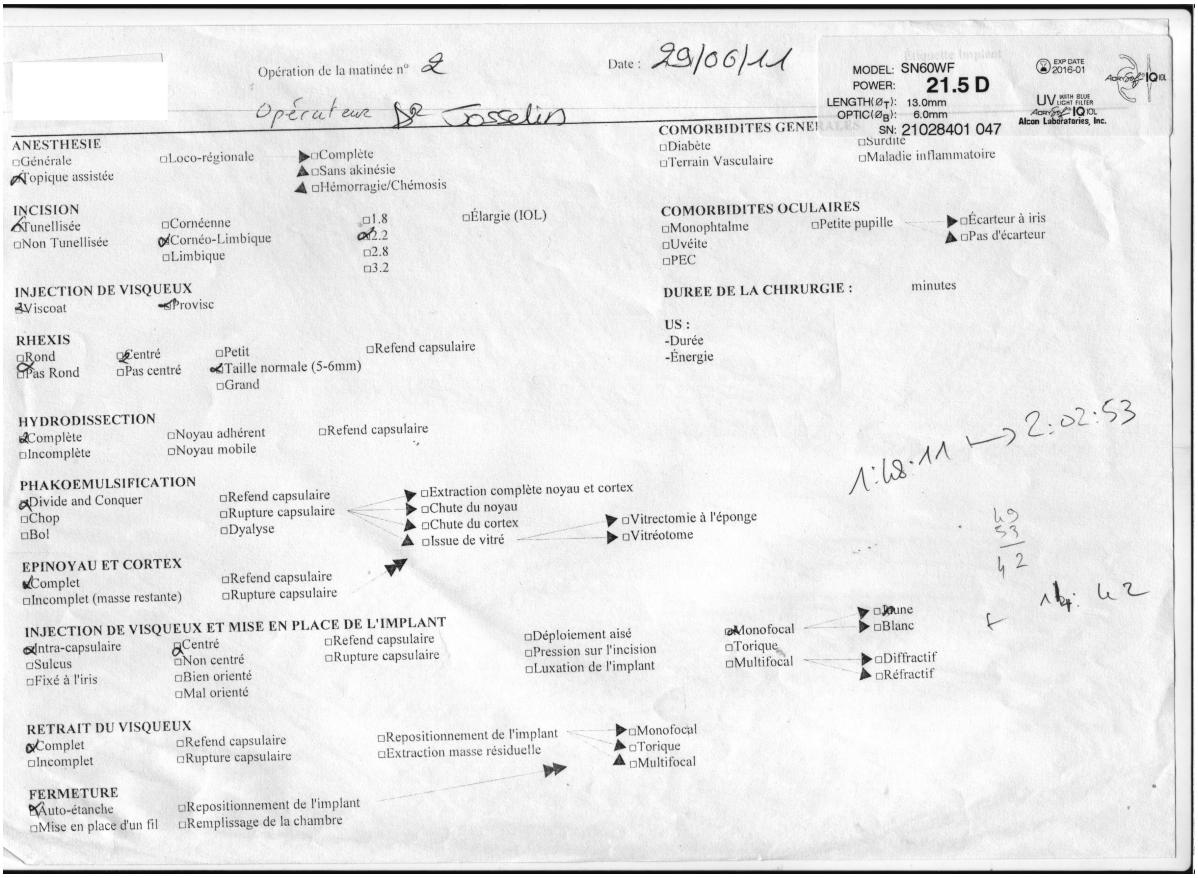


Figure B.5 — Description de l'opération de la cataracte établie par le Dr. Josselin